NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A WARTIME SUSTAINABILITY MODEL
FOR A SMALL COUNTRY

by

Lim Hung Heng

March, 1991

Thesis Advisor: Melvin B. Kline
Co-Advisor: Alan W. McMasters

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b. OFFICE SYMBOL (If applicable) 55	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000	7b. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS Program Element No. / Project No. / Task No / Work Unit Accession Number

11. TITLE (Include Security Classification)
A WARTIME SUSTAINABILITY MODEL FOR A SMALL COUNTRY

12. PERSONAL AUTHOR(S)
LIM, HUNG-HENG

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (year, month, day)	15. PAGE COUNT
Master's Thesis	From To	March 1991	191

16. SUPPLEMENTARY NOTATION
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17. COSATI CODES			18. SUBJECT TERMS (continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUBGROUP	Wartime environment, Sustainability, Operational Availability, Not-Mission-Capable systems, Repair policies, Cannibalization, OPUS-8, Logistics Support, Spares.

19. ABSTRACT (continue on reverse if necessary and identify by block number)

The transient behavior of a generic military capability under wartime environment was analyzed and, under certain assumptioms, a Wartime Sustainability Model (WSM) was developed analytically using various stochastic and inventory techniques. A simulation of the WSM was also developed to incorporate variations for repair policies such as repair prioritization and limited repair capability. These variations are extremely difficult to model analytically. The adequacy of these two models was verified using a numerical example. Finally, the feasibility of using OPUS-8, a steady-state spares optimization model developed by Systecon AB, Stockholm, Sweden, as an approximation to the anlytical version of the WSM for the case of no repair capability was also investigated.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS REPORT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL Melvin B. Kline	22b. TELEPHONE (Include Area code) (408) 624-9541 22c. OFFICE SYMBOL AS/Kx

DD FORM 1473, 84 MAR   83 APR edition may be used until exhausted. All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

Approved for public release; distribution is unlimited.

A Wartime Sustainability Model
For A Small Country
by

Lim Hung Heng
Major (Notional), Singapore Armed Forces
B.S., University of Manchester (U.K.), 1984

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
March 1991

Author: ______________________
Lim Hung Heng

Approved by: ______________________
Melvin B. Kline, Thesis Advisor

____
Alan W. McMasters, Co-Advisor

____
Michael P. Bailey, Second Reader

____
Peter Purdue, Chairman
Department of Operations Research

ABSTRACT

The transient behavior of a generic military capability under wartime environment was analyzed and, under certain assumptions, a Wartime Sustainability Model (WSM) was developed analytically using various stochastic and inventory techniques. A simulation of the WSM was also developed to incorporate variations for repair policies such as repair prioritization and limited repair capability. These variations are extremely difficult to model analytically. The adequacy of these two models was verified using a numerical example. Finally, the feasibility of using OPUS-8, a steady-state spares optimization model developed by Systecon AB, Stockholm, Sweden, as an approximation to the analytical version of the WSM for the case of no repair capability was also investigated.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

I would like to express my gratitude to all the persons and institutions that have given me this unique opportunity to widening my horizon at the Naval Postgraduate School. I am particularly grateful to the Defence Material Organization of the Singapore Defence Ministry for this sponsorship.

My sincere thanks to my advisor, Professor Emeritus M. B. Kline who has given me great insights in the execution of this thesis. I am equally indebted to the guidance of my co-advisor Professor A. W. McMasters and second reader, Professor M. P. Bailey.

Last but not least, my heart-felt appreciation to my wife, Violet, who always supported and encouraged me especially during those times when the going was tough.

I. INTRODUCTION

A. MOTIVATION FOR THE THESIS

The Ministry of Defence (MINDEF) of the Republic of Singapore recognized the need to optimize its investment in spares to maintain the desired level of readiness of its weapon systems and, therefore, has purchased under license a software package called "OPUS" [Ref. 1]. Installed in 1985 and updated to the present version, OPUS-8 has since been used to perform steady-state optimal allocation of spares for many types of weapon systems. This has led to a better understanding of the kind of effects a given allocation of spares have on Operational Availability *(Ao),* a widely accepted system-oriented steady-state Measure of Effectiveness *(MOE)* used for peacetime deployment.

However, during a wartime period, utilization of a military capability becomes highly dynamic and this can result in inevitable interruptions and delays in the logistics support for the military capability. OPUS-8 does not address this problem directly because of its steady-state assumptions. Moreover, the operational planners and users are more comfortable with the use of mission-oriented MOEs to define the system's readiness during such dynamic periods.

For these reasons, both the Joint Logistics and Operations Analysis departments of MINDEF have specifically expressed the need to establish sound

policies and approaches to relate spares and related logistics resources to a mission-oriented *MOE* under a dynamic wartime environment.

Existing sustainability models such as Dyna-METRIC [Ref. 2] and the Aircraft Sustainability Model [Ref. 3] have been developed with the specific purpose of studying the effects of spares on the mission-readiness of an aircraft squadron as a detachment unit in a remote location where logistics support is limited. Although the basics of these models are readily accessible, the use of these models are restricted to US and NATO military agencies. Hence, the author was motivated to investigate the concepts and algorithms of these models with the aim of developing a specific Wartime Sustainability Model (WSM) for use in MINDEF.

B. GOALS

A goal of the thesis was to formulate relevant policies, analytical assumptions and rationales for the development of an analytical model for the WSM.

Another goal was to develop a simulation version for the WSM to study the effects of policies such as cannibalization, repair prioritization and limited repair resources on the sustainability of a military capability during an anticipated wartime period. Such policies are extremely difficult to model analytically.

Both models are verified with a numerical scenario and the results contribute towards a better understanding of the transient behavior of a military capability under a dynamic combat environment.

Another objective of this thesis was to use the existing OPUS-8, a steady-state model, as an approximation to the proposed WSM under the special case of no repair capability.

C. PREVIEW

The basic layout of this thesis and the relationships between the chapters and the appendices are illustrated by Figure 1-1.

Chapter I introduces the reader to the motivation and goals of this thesis. Chapter II establishes a scope or framework for the development of the WSM in a systematic and effective manner. Chapter III shows the development of the analytical model for the WSM under specific stochastic assumptions. Chapter IV describes the simulation version for the WSM under different situations using a fourth generation simulation language, MODSIM II [Ref. 4], to implement the simulation. The MODSIM Program listing of the simulation model is provided in Appendix A. Chapter V explores the possibility of using the current features of OPUS-8 to approximate the WSM. Chapter VI contains many numerical results and graphs based on an example scenario, and analyses of these results are provided. For efficient computation, the exact analytical expressions derived in Chapter III were coded into a computer program using PC-MATLAB [Ref. 5] syntax, providing numerical solutions to the required numerical examples. This PC-MATLAB program listing is given in Appendix B. Examples of computer outputs from the analytical model, the simulation model and the OPUS-8 Approximation are also

provided in Appendices C, D and E, respectively. Chapter VII summaries the thesis effort and provides conclusions and recommendations for further model development.

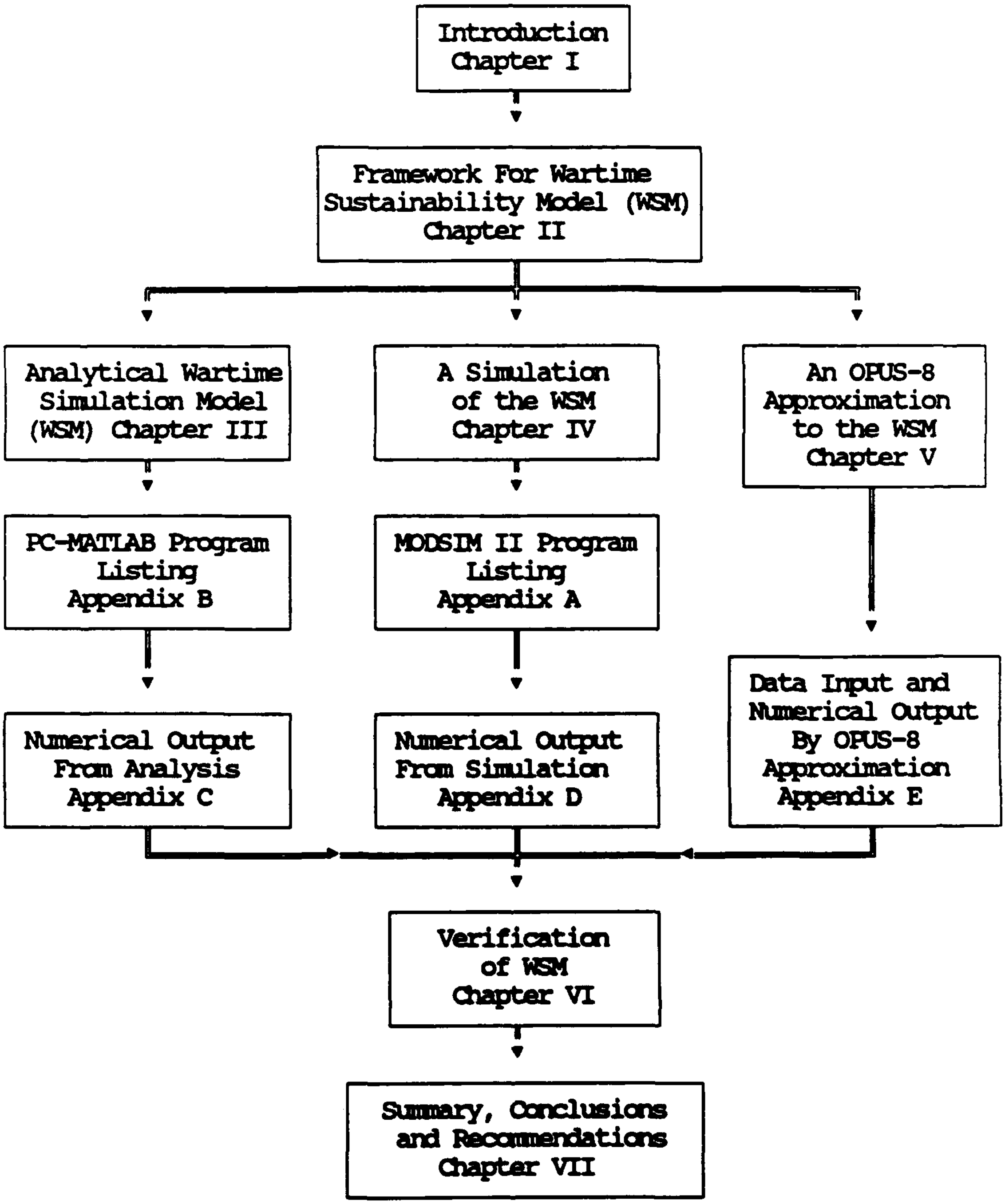

FIGURE 1-1. THESIS LAYOUT

II. FRAMEWORK FOR A WARTIME SUSTAINABILITY MODEL (WSM)

A. SUSTAINABILITY DEFINITION

Sustaining a military capability in a wartime environment is a complex combination of many military aspects such as manpower, tactics and logistics support.

For the purpose of this thesis, the term "sustainability" applies only to the prediction of the effects of logistics support policies on the ability of the military to sustain its capability during an anticipated wartime period. For example, the military capability can be a squadron of aircraft deployed at a particular base supported by many logistics resources.

Because of the short period of the time allocated for this thesis, the design of the WSM is restricted to being an assessment tool for analyzing the effects of a prespecified allocations of spares and repair resources on the mission readiness of the military capability. In other words, the WSM to be developed in this thesis is not an optimization tool capable of recommending further requisition of an optimal number of spares and repair resources when the initial allocation is determined to be inadequate. Further development of the WSM is envisioned when the author returns to Singapore.

Policies and Measures of Effectiveness *(MOEs)* which have influences on the design and development of the WSM are discussed in the following sections.

B. POLICIES

There are many problems involved in trying to establish sound policies to guide the determination of wartime logistics support. One of them is the lack of a common terminology for this area which leads to much confusion and misunderstanding between the operational planners and the logisticians who are responsible for implementing these policies. Another is that even when it is accepted that operational requirements must be defined first before logistics resources can be determined, these requirements are often not specific enough to be translated into reasonable logistics objectives.

Research carried out by the Logistics Management Institute of U.S.A. emphasized that proper determination of war reserve spares requires clear and sound policies which must be endorsed by both the users and the logisticians [Ref. 6]. The same reasoning should apply in the context of MINDEF and the author feels that the following policy issues should be addressed as part of developing an acceptable WSM.

1. War and Mobilization Plan (WMP).

Each major military capability in the Singapore Armed Forces (SAF) has a War and Mobilization Plan formulated by the MINDEF strategic planners. In the context of the WSM, the WMP should address the following issues:

a. the total wartime period which includes the pre-tension period, the surge period and post-tension period;

b. the utilization of this capability during these periods;

c. war attrition; and

d. the minimum critical numbers for the various facets of the capability to sustain the wartime period.

An example of what might be included in the WMP is the utilization profile shown in Figure 2-1 for a squadron of aircraft during an anticipated wartime period.

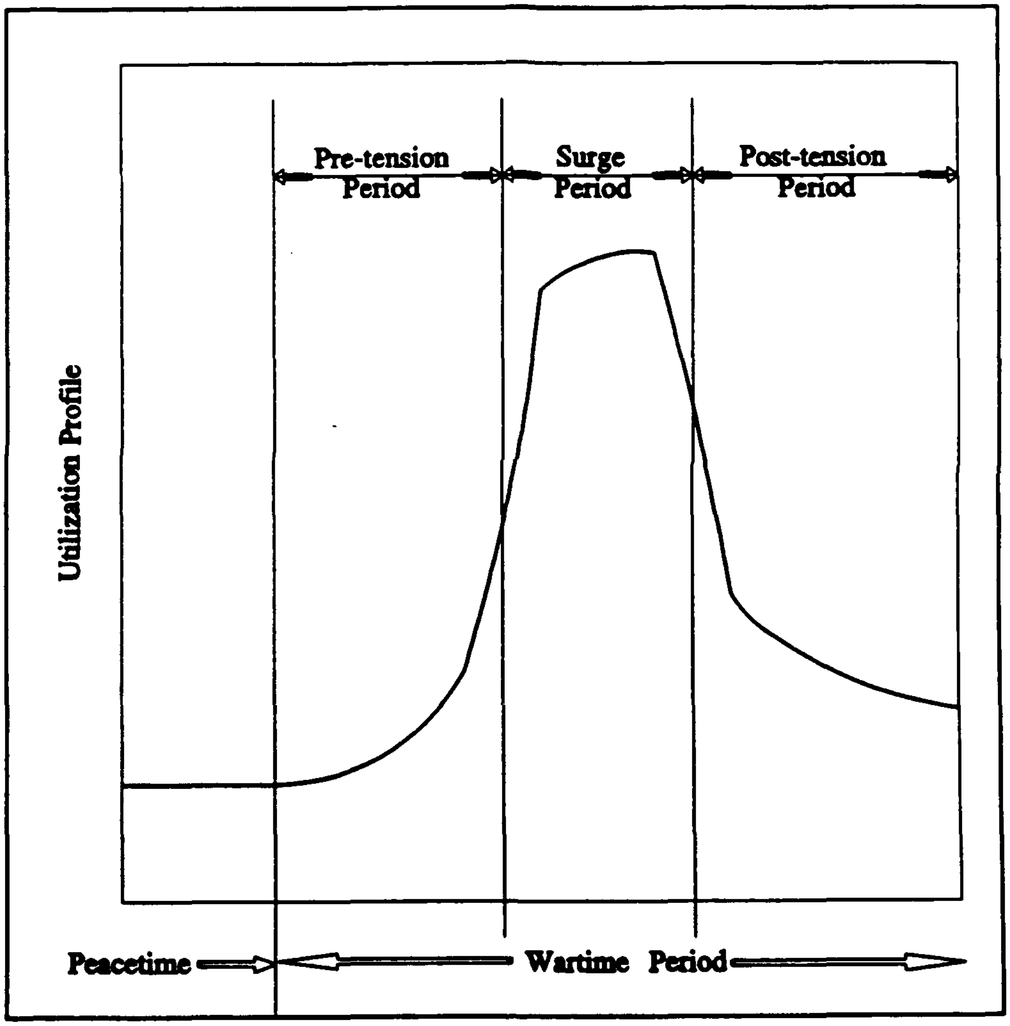

FIGURE 2-1. TYPICAL WARTIME UTILIZATION PROFILE

It is assumed in this thesis that in the earlier stages of planning to acquire the assets needed to achieve the military capability, war attrition analysis has already been performed to ascertain the minimum critical mass of the assets required to sustain an anticipated war. As a consequence, the effects of attrition will not included in the design of the WSM.

2. Optimization of Investment In Spares

Given the limited allocated funds for defense, the need to maximize a military capability for a given dollar amount cannot be overemphasized. Although not specifically addressed in this thesis, there is a need to buy the correct assortment of spares and repair resources for a particular asset investment to achieve the maximum MOEs with a limited amount of funds. The proposed WSM can be used in a limited way to answer this policy. But, when expanded into an optimization tool, the WSM will be capable of fully addressing this issue.

3. The Inadequacy of A Minimum Buy Policy

In the past, it has been common for the SAF to buy war-reserve spares based on a minimum-buy policy. However, such a minimum quantity was often bought based on the suppliers' recommendations. Another method was to buy spares for each item to meet a specified service level based on the assumption of Poisson distribution for the demands for spares over the specified period of conflict. Both approaches disregard the need for optimization with respect to any *MOEs*. In fact, the minimum-buy policy prevents resources from being available for a desired

MOE. This has been shown to be true by a study carried out under the supervision of the Joint Logistics Department of MINDEF which highlighted the urgency to minimize the excessive amount of unused spares in the SAF inventory system. Also, the minimum-buy policy fails to take into account intense fluctuations of demands during a wartime period and therefore is not able to accommodate the demands for spares and repair resources during a surge period. The proposed WSM is a more effective and systematic approach for surge protection during a wartime period.

4. Cannibalization Policy.

During peacetime deployment it makes sense to discourage cannibalization of systems for spare parts since this practice can create havoc within any logistics management accounting system. However, in time of war where the chief objective is to maximize the utilization of all available systems to accomplish a mission, studies have shown that cannibalization does improve operational availability *(Ao).* However one study recommended a policy to control cannibalization by setting an upper limit on the number of systems to be cannibalized [Ref. 6]. This study also showed that a correct choice of this limit on cannibalization can maximize certain *MOEs.* The effects of cannibalization are considered in the development of the WSM.

5. Repair Prioritization Policies

An appropriate choice of a repair prioritization policy is important when repair resources are limited, since a decision to repair one item also means a decision not to repair another when both are competing for the same repair resources. Repair priority policies such as First Come First Served *(FCFS)* and Least Availability Item First *(LAIF)* are commonly adopted. Of course, other policies may be more suitable under certain circumstances.

With the above in mind, the properly defined WMP enables the operational planners to develop relevant Measures of Effectiveness *(MOEs)* which can be used as design criteria for the development of the WSM. Details of such *MOEs* are discussed in section II-C.

C. MEASURES OF EFFECTIVENESS (MOEs)

The WMP provides a basis for determining the relevant Measures of Effectiveness *(MOEs)* which can serve as design goals for the development of the WSM. *MOEs* can be further classified into System-oriented *MOEs* and Mission-oriented *MOEs* and their relationships are depicted in Figure 2-2. The dependence of these *MOEs* on the performance characteristics of the military asset is also shown. However, the effects of performance attributes on a military asset will not be considered in the WSM since this is beyond the scope of this thesis.

Mission-oriented
Measures Of Effectiveness ⇒ Examples
ENMCS, P[ENMCS≤NMCS]
⇑ ⇑ Kill Probability,
Attrition rate.

MISSION EFFECTIVENESS ANALYSIS

⇑ ⇑
System-oriented
Measures Of Effectiveness ⇒ Examples
Ao(t)(Wartime),
⇑ ⇑ Ao (Peacetime),
MTTR, MLDT, MTBF.

SYSTEM EFFECTIVENESS ANALYSIS

⇑ ⇑
Performance-oriented
Measures ⇒ Examples
Altitude, Range,
⇑ ⇑ Speed, Radar Cross
Section.

PERFORMANCE ANALYSIS

FIGURE 2-2 : RELATIONSHIPS BETWEEN THE MOEs FOR A MILITARY CAPABILITY

1. Mission-oriented MOEs (MMOEs)

The proposed *MMOE* for the WSM is *NMCS* where

NMCS = the maximum allowable number of Not-Mission-Capable Systems that can be tolerated without reducing the military capability during an anticipated wartime period.

NMCS a mission objective which should be specified by the operational planners and its value should set to at least a 95% upper confidence limit. It is important to emphasize that the proposed WSM presumes that the user had undertaken comprehensive studies (analyses or wargaming) to obtain the *NMCS.* Therefore, this objective must be in the WMP for the desired military capability. The WSM computes the Expected number of Not-Mission-Capable Systems *(ENMCS)* based on a given allocation of spares and repair resources.

The military capability is not severely downgraded as long as *ENMCS* is less than *NMCS.* Reference 6 preferred this *MMOE* to avoid possible statistical confusion associated with confidence-level oriented objectives.

Another *MMOE* candidate is the Confidence Level of having not more than a specified number of failed systems throughout the anticipated wartime period. It can be formulated as *Probability [ENMCS < = NMCS].*

1. System-oriented MOEs (SMOEs)

Many steady-state spares models including OPUS-8 use Operational Availability *(Ao)* as a key *SMOE.* However, in a dynamic environment, *Ao* becomes *Ao(t)* which is time-dependent. *Ao(t)* is computed based on the ratio of the expected number of Mission-Capable Systems *(MCS)* and the total number of deployed systems. It is shown in Chapter III that *ENMCS* and *Ao(t)* have a direct relationship.

The main focus of the thesis is on the effects of spares on both *Ao(t)* and *ENMCS*. In Chapter III, other two *SMOEs*, the expected number of demands for spares and the expected number of backorders for each Line Replaceable Unit *(LRU)*, are shown to be basic building blocks of the WSM for achieving desired *Ao(t)* and *ENMCS* levels.

D. SYSTEM STRUCTURE AND LOGISTICS SUPPORT

In order to be operationally ready, the deployed systems have to be supported by many logistics resources. A typical organization for logistics support of a major system having multiple indenture-levels is depicted in Figure 2-3. The base repair and supply facilities are shaded to emphasize the proximity of these locations to the deployment sites.

When a Line Replaceable Unit *(LRU)* of the system breaks down at the base, the system will enter a base repair network which has support capabilities (men, machines, spares and supplies) to remove and replace any system's faulty *LRU* with a good one. Depending on the nature of the failure, the failed LRU is sent either to the base repair facility, the intermediate repair facility or the depot repair facility for the appropriate repairs.

Demands for spares of an *LRU* are intense during a wartime period and therefore the system requires a sufficient number of spares of each *LRU* so that the system will never be unavailable due to lack of spares.

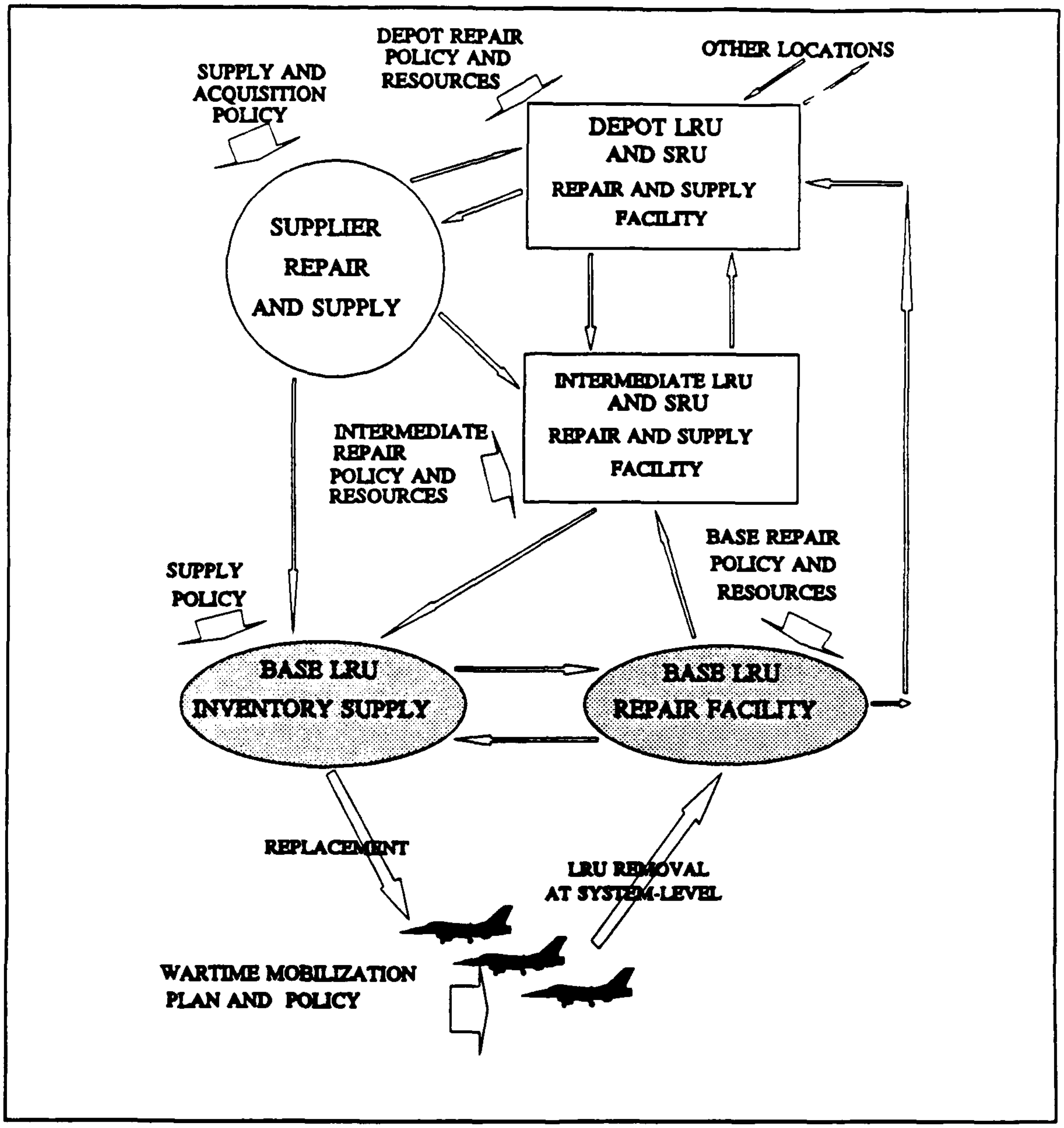

FIGURE 2-3 : A TYPICAL LOGISTICS SUPPOPT ORGANIZATION

III. AN ANALYTICAL WARTIME SUSTAINABILITY MODEL (WSM)

This chapter first describes the assumptions required by the WSM analytical model to obtain the relevant exact analytical expressions for the *SMOEs* and *MMOEs*. In the development of the formulae, proofs for well-known theorems will only be referenced to the appropriate sources.

A. ANALYTICAL MODEL ASSUMPTIONS

The following assumptions are made to enable the development of an analytical model for the WSM:

a. Failures of the Line Replaceable Units *(LRUs)* are generated at the base level. These failures are repairable at the base with a certain probability, otherwise they will be shipped to the depot for repair. Also, it is assumed that there will be no preventive maintenance in a wartime environment.

b. All faulty *LRUs* shipped to the depot can be repaired (i.e., there are no condemnations or attritions at the depot).

c. The distribution of the interarrival times of any *LRU* failure follows either a nonhomogeneous Poisson distribution or a nonhomogeneous compound Poisson distribution.

d. The failure of one type of *LRU* is statistically independent of those which occur for any other type of *LRUs*.

e. The repair times and transportation times are statistically independent of one another. These times can be stationary or non-stationary.

f. Repair resources are assumed to be unlimited.

B. FAILURE ARRIVAL RATES

Let's look at a particular *LRU* of the system. Since the system has a varying utilization rate during an anticipated wartime period, the arrival rate of the *LRU* failure, $D(t)$, at an instant in time has the following formula:

$$D(t) = f \cdot q \cdot u(t) \cdot ns(t), \tag{3.1}$$

where

f = the basic item failure rate of the *LRU*. Here, interarrival times are also assumed to follow an e::ponential distribution (i.e., it is constant).

q = quantity of the *LRU* found in each system.

$u(t)$ = the utilization rate per day of all the deployed systems at time t. Its value varies with time based on the WPM as specified by the operational planners. For example, if the systems are required to operate for 12 hours a day at time t, then $u(t)$ has a value of 0.5.

$ns(t)$ = the number of available systems at time t. This also changes with time since some systems may become unavailable due to the lack of a spare.

C. DEMANDS FOR SPARES

In this section we present the derivations of the exact analytical expressions for the distribution of the number of failed units of a particular *LRU* being held up in repair at a given time t. This number of failed units generated demands for spares for the same *LRU* which were hopefully satisfied by the inventory in stock. Systems become *Not-Mission-Capable* when there is inadequate inventory.

1. Nonhomogeneous Poisson Assumption

a. *The Distribution of LRU Failure Arrivals*

Let $\{N(t),\ t \geq 0\}$ be the counting process of a particular *LRU's* failure arrivals which follows a nonhomogeneous Poisson process. Also assume that there are no arrivals before time $t=0$ (i.e., the arrival process starts empty; that is, $N(0)=0$). The expected value, denoted by $M(t)$, for the process $N(t)$ is

$$M(t) = E[N(t)] = \int_0^t D(s)\, ds \ . \tag{3.2}$$

b. *The Distribution of the Repair Process*

Corresponding to an *LRU* failure arrival that occurs at time s, let the repair time random variable denoted by Y, have a distribution function $G(s,\ s+y)$ which is dependent on s. The actual repair times y are also assumed to be independent.

c. *The Distribution of Failed Units of an LRU Still in Repair*

We wish next to determine the number of arrivals still in repair, denoted by $Z(t)$, for the case of unlimited repair resources.

Suppose the Poisson arrivals have a constant arrival rate D and undergo repair with repair times independent and identically distributed *(iid)* from a stationary distribution F. Ross [Ref. 7], Example 4b, p.237, showed that $Z(t)$ also has a Poisson distribution with expected value of

$$E[Z(t)] = \Lambda(t) = \int_0^t D[1 - F(s)]\, ds \,. \tag{3.3}$$

Takacs [Ref. 8], p.160, further showed that as $t => \infty$, the limiting distribution of the number of arrivals still in repair follows a Poisson distribution with mean

$$\lim_{t \to \infty} \Lambda(t) = D\, E[Y] \,. \tag{3.4}$$

so long as the limit is finite (i.e., $1/D = E[Y]$).

This is widely known as Palm's Theorem.

Palm's Theorem was extended further by Hillestad and Carillo [Ref. 9] to handle a particular *LRU* having a nonhomogeneous Poisson failure and undergoing a repair process. The assumption that the failure arrivals and repair times are independent of one another is needed. Under these conditions, $Z(t)$ also has a nonhomogeneous Poisson distribution with the expected value, denoted by $\Lambda(t)$, as

$$\Lambda(t) = \int_0^t [1 - G(s,t)] D(s)\, ds \tag{3.5}$$

where

s = time when the repair was initiated, and

t = time of interest.

Finally, the distribution of $Z(t)$ is given by

$$P[Z(t) = k] = e^{-\Lambda(t)} \frac{\Lambda(t)^k}{k!} \tag{3.6}$$

Instead of $P[Z(t)=k]$, the notation $P[k\,;\Lambda(t)]$ will be used from now on where $\Lambda(t)$ represents the expected number of failed units of *a particular LRU* still in repair. $\Lambda(t)$ can also be interpreted as the expected number of outstanding demands for spares for that same *LRU*. If there are enough spares in stock, then these demands do not cause a system to be *Not-Mission-Capable*.

Since $Z(t)$ is a Poisson random variable at time t, its variance, $Var[Z(t)]$, is equal to $E[Z(t)]$(i.e., $\Lambda(t)$). The ease of computation resulted from this closed-form expression will be demonstrated in Chapter VI.

2. Nonhomogeneous Compound Poisson Assumption

a. *The Distribution of LRU Failure Arrivals*

Intense failure arrival rates of a particular *LRU* during a wartime scenario can be expected. Also it is reasonable to expect an arrival to consist of a batch of units of the *LRU* instead of just one.

W_n is defined to be the batch size at the nth arrival of the Poisson process $N(t)$. It is assumed that the W_n, $n = 0, 1,...$ are *iid* random variables having a common compounding distribution $\{ C_j = P[W=j], j = 0, 1,... \}$, with its expected

value defined as $E[W]$. Assume further that the family $\{ W_n \}$ is independent of the arrival process.

With the above conditions, we define $\{X(t),\ t \geq 0\}$ as the resulting counting process of the number of failure arrivals for a particular *LRU*, where

$$X(t) = \sum_{n=1}^{N(t)} W_n \tag{3.7}$$

Ross [Ref. 10], p.49, has shown that $X(t)$ is a compound Poisson process having the following expressions for its mean and variance,

$$E[X(t)] = M(t)E[W] \tag{3.8}$$

and

$$Var[X(t)] = M(t)E[W^2]\ . \tag{3.9}$$

b. *The Distribution of Failed Units of an LRU Still in Repair*

Again, we are interested in $Z(t)$, the number of failed units of a particular *LRU* still in repair. If the failed units of the *LRU* in each batch have independent repair times from a distribution G (this can be treated as a special case of an $M/G/\infty$ queue with batch arrivals and general service), then the resulting $Z(t)$ also follows a compound Poisson process with

$$P[Z(t) = k] = \sum_{j=0}^{\infty} \frac{C_k^{(j)}\, e^{-\Lambda(t)}\, [\Lambda(t)]^j}{j!} \quad for\ j = 0, 1, \ldots \tag{3.10}$$

where $\{C_k^{(j)}\}$ is the j-fold convolution of the compounding distribution $\{C_k\}$ with itself. The proof for Equation 3.10 is given by Feeney and Sherbrooke [Ref. 11]. Therefore *Z(t)*, the number of failed units of LRU still in repair, is nonhomogeneous Poisson with compounding distribution $\{ C_j, j = 0, 1, 2, ... \}$. Also *Z(t)* may not have a closed form expression due to the convoluted term, $\{C_k^{(j)}\}$, in Equation 3.10. However it is shown in Feller [Ref. 12], p.291, that if *W* is assumed to follow a logarithmic probability function with probability density function *(PDF)* of

$$P[W=j] = -j[1 - a] / [j\log(a)], \text{ for } 0 < a < 1; j=1,2,... \tag{3.11}$$

and its the expected number given as

$$E[W] = - [1 - a] / a\log(a) \tag{3.12}$$

where

$$1/a = E[W^2]/E[W] = VMR,$$

and *VMR* is known as the Variance-to-Mean Ratio of the distribution with a constant value always greater than 1, then the distribution of *X(t)* follows a negative binomial distribution, having parameters $\{ a, R(t) \}$ where

$$R(t) = - M(t) / \ln(a) . \tag{3.13}$$

Thus, according to Feller [Ref. 12], *Z(t)* will also have negative binomial distribution with parameters

$\{ a, R(t) \}$ where

$$R(t) = -\Lambda(t) / \ln(a) . \tag{3.14}$$

Also, its *PDF* is

$$P[Z(t)=k] = \binom{R(t)+k-1}{k} a^{R(t)} (1-a)^{k} \quad for\ k = 0, 1, \ldots \tag{3.15}$$

D. MULTIPLE-INDENTURE PIPELINES

In the previous section, exact analytical expressions for the number of demands for spares for each *LRU* have been derived. Therefore, the number of units of the *LRU* in repair at the repair base, repair depot as well as in transit between the base and the depot can be computed using Equations 3.6 and 3.15. From here onwards, the term *Pipeline* is used to replace the phrase "expected number of failed units in repair and/or in transit of a particular *LRU*". For example, the expected units in repair at the repair base will be referred to as the *Base Pipeline*. Pipeline can also be interpreted as the expected number of demands for spares for that *LRU*.

This section further develops the expressions for the pipelines to accommodate a system with more than one level of item breakdowns (i.e., multiple-indenture system).

1. A Case of Two-Indenture System

A system can be broken down into many indenture levels depending on the complexity of the system design. For the purposes of this thesis, two levels of item breakdowns are analyzed (i.e., a two-indenture system).

The terminology of Line Replaceable Units *(LRUs)* is used for the major items whose failures cause the system to be down if no spare *LRU* is available. Shop Replaceable Units *(SRUs)* are second-level items which make up the *LRUs*. The failure of an *SRU* creates a "hole" in an *LRU* which may cause a failure of the *LRU*. Therefore, a failed *SRU* may indirectly lead to a system being *Not-Mission-Capable*.

Muckstadt [Ref. 13] has shown that for steady-state cases, Little's Formula, a classical queuing result, can be adapted to analyze each repair pipeline given that an *LRU* has to wait for *SRUs* for its repair. The approach used in the WSM is similar except that steady-state assumptions can now be relaxed.

Let $Q_i(t)$ represents the quantity of LRU_i waiting for *SRUs* at time t and let $EQ_i(t)$ be its expected value. This quantity is also equivalent to the number of LRU_i being held up at the repair facility due to the lack of *SRUs*. Because of this, additional demands for spares are generated. For ease of explanation, the following analysis considers only a particular base but the results are applicable to any base or depot. Based on the nonhomogeneous Poisson assumptions made in subsection III-C1, let $\Lambda^b{}_i'(t)$ be the *Base Repair Pipeline* for LRU_i. Then according to Muckstadt

[Ref. 13], the *Base Repair Pipeline* of LRU_i, when taking into account the pipeline due to lack of *SRUs*, is revised to become

$$\Lambda_i^b(t) = \Lambda_i^{b'}(t) + EQ_i(t) \ . \tag{3.16}$$

The analysis carried out in this subsection is only valid for *LRUs* which undergo repair. When there is no repair allowed for the LRU, then there will not be any shortages due to waiting for *SRUs*. The following analyses also assume the independence between the *LRU* and its *SRUs'* demand distributions even though it is more realistic that the failure of an *LRU* is due to its *SRUs'* failures.

a. *Cannibalization Policy*

We now look at how cannibalization can affect the computation of $EQ_i(t)$. First, shortages of *SRUs* can be consolidated into the smallest possible number of LRU_i. This is accomplished by using a serviceable *SRU* of a failed *LRU* to repair another *LRU* which requires the serviceable *SRU*. Let $P^j(n,t)$ be the probability that the shortages of the SRU_j are less than or equal to a quantity n at time t. Then $P^j(n,t)$ is

$$P^j(n,t) = \sum_{m=0}^{S^j(t)+n} [\ P(\ \textit{number of } SRU_j \ \textit{failures}=m \ \textit{at time } t\ )] \tag{3.17}$$

where $S^j(t)$ represents the inventory level of SRU_j at the repair site at time t.

The exact form of the probability $P(.)$ in Equation 3.17 depends on whether the demands for SRU_j are Poisson or Compound Poisson. Now let $P_i(n,t)$ be the probability that the number of LRU_i waiting for its $SRUs$ to be repaired is less than or equal to n. Failure independence among the $SRUs$ and among the $LRUs$ are assumed. Assume that the LRU_i is made up of different types of $SRUs$ connected in series, then $P_i(n,t)$ can be obtained by the multiplication of all its $SRUs$' $P^j(n,t)$. This is an application of reliability theory for a serial model. Therefore $P_i(n,t)$ is

$$P_i(n,t) = \prod_{j=1}^{J} P^j(n,t) \ , \tag{3.18}$$

assuming that there are a total of J different $SRUs$ in LRU_i .

Then the expected number of unavailable LRU_i due to shortages of $SRUs$ can be computed from the conditional expectation of the above distribution

$$EQ_i(t) = \sum_{n=0}^{K} [1 - P_i(n,t)] \ , \tag{3.19}$$

where K is the total number of SRUs available which is equivalent to

K $=$ $ns(t)$ J + $S^j(t)$, and

$S^j(t)$ = the inventory level of SRU_j at time t.

From its distribution function given in Equation 3.18, the probability density function *(PDF)* of the number of LRU_i that have to wait for $SRUs$ at time t can be computed as

$$PDF_i(n,t) = P_i(n,t) - P_i(n-1,t). \tag{3.20}$$

In other words, $PDF_i(n,t)$ is the probability of having exactly n units of LRU_i, each waiting for at least one of its *SRUs* to be repaired.

b. No Cannibalization

Reference 13 also showed that the computation of $EQ_i(t)$ when no cannibalization of *SRUs* is allowed can be developed as follows. The probability that LRU_i is short of its SRU_j, given that there are $B(t)$ *Backorders* of SRU_j at time t can be written as

$B(t)/[ns(t) + S^j(t)]$ if we assume that there is only one of each type of SRU_j in LRU_i. The formulation is true only if:

(a) it is finite,

(b) failure times are exponential , and

(c) there is no information about $[0, t]$.

The probability that LRU_i has a shortage of SRU_j is computed by summing over all possible $B(t) = k - ns(t) + S^j(t)$ multiplied by the probability that they occur, which is

$$\sum_{k=ns(t)+S^j(t)+1}^{\infty} \frac{[k - ns(t)+S^j(t)]}{ns(t)+S^j(t)} P[k\ ;\ \lambda^j(t)] = \frac{EB^j(t)}{ns(t)+S^j(t)}\ . \tag{3.21}$$

where

$\lambda^j(t)$ = expected number of demands for SRU_j at time t,

and because the numerator

$$\sum_{k=ns(t)+S^j(t)+1}^{\infty} [k - ns(t)+S^j(t)]P[k \; ; \; \Lambda^j(t)]$$

is equivalent to $EB^j(t)$, the expected number of *Backorders* of SRU_j at time t. This equivalence will become evident later in section III-F1a. Therefore, the probability that LRU_i does not have any shortage of SRU_j is the complement of the distribution shown in Equation 3.21.

Assuming that all failures are independent and all the *SRUs* are connected in series to form LRU_i, the probability that LRU_i does not have any shortage of all its *SRUs* is just

$$\prod_{j=1}^{J} [\, 1 - \frac{EB^j(t)}{ns(t)+S^j(t)}] = P[\; LRU_i \textit{ is Mission-Capable} \;] \quad . \tag{3.22}$$

Again, the complement of the distribution in Equation 3.22 is equivalent to the probability that LRU_i has shortage of all its *SRUs*. If we assume that there is only one unit of each SRU_j found in LRU_i then the expected number of LRU_i having shortages of *SRUs* is given by

$$EQ_i(t) = [(ns(t)+S^j(t)].[1 - \prod_{j=1}^{J}(1 - \frac{EB^j(t)}{(ns(t)+S^j(t)})] \quad . \tag{3.23}$$

E. MULTIPLE-ECHELON PIPELINES

In this section we examine the logistics support for the entire deployment of systems. A *Total Pipeline* is obtained by combining the *Pipelines* at the repair facilities and in-transit between these facilities.

1. A Case of Two-Echelon Logistics Support

In this analysis, it is assumed that there is only one base repair facility and one depot repair facility. Let a failed *LRU* be repairable at the depot level with fixed probability commonly known as Not-Repairable-This-Station *(NRTS)* and at the base level with probability *(1 - NRTS)*, independent of where other failed *LRUs* are repaired. Then, the single nonhomogeneous Poisson arrival stream with mean value function *M(t)* becomes two independent nonhomogeneous Poisson streams [Ref. 9] with mean value functions expressed as

$$\left.\begin{array}{ll} NRTS\ M(t) & \text{for the depot,} \\ (1 - NRTS)\ M(t) & \text{for the base.} \end{array}\right\} \qquad (3.24)$$

When the *Repair Pipeline* at each repair facility is assumed to follow a nonhomogeneous Poisson process, it will be denoted by

$\Lambda^b(t)$ = *Base Repair Pipeline*
= expected quantity of *LRUs* being repaired at the base *(including the LRUs waiting for SRUs)* at time *t*,

$\Lambda^d(t)$ = *Depot Repair Pipeline*
= expected quantity of *LRUs* in repair at the depot *(including the LRUs waiting for SRUs)* at time *t*

and the formulae for $\Lambda(t)$ can be obtained from Equations 3.16 and 3.24.

a. Total Repair Pipeline

The following analysis reveals that there are two cases for which the *Total Repair Pipeline* can take on exact analytical results.

(1) Nonhomogeneous Poisson Pipelines

When the *Base Repair Pipeline* and the *Depot Repair Pipeline* are nonhomogeneous Poisson processes and are shown to be independent (allowing Equation 3.24), then the *Total Repair Pipeline* is also a nonhomogeneous Poisson properties with its mean value equal to the sum of the values of the *Base Repair Pipeline* and the *Depot Repair Pipeline.* The proof for this has been provided in Ross [Ref. 7]. Therefore, the mean of the *Total Repair Pipeline,* $\Lambda_T(t)$, for a particular *LRU* at any time *t* is just the summation

$$\Lambda_T(t) = \Lambda^b(t) + \Lambda^d(t)\,. \tag{3.25}$$

Figure 3-1 illustrates Equation 3.25 for the data from the numerical example in Chapter VI.

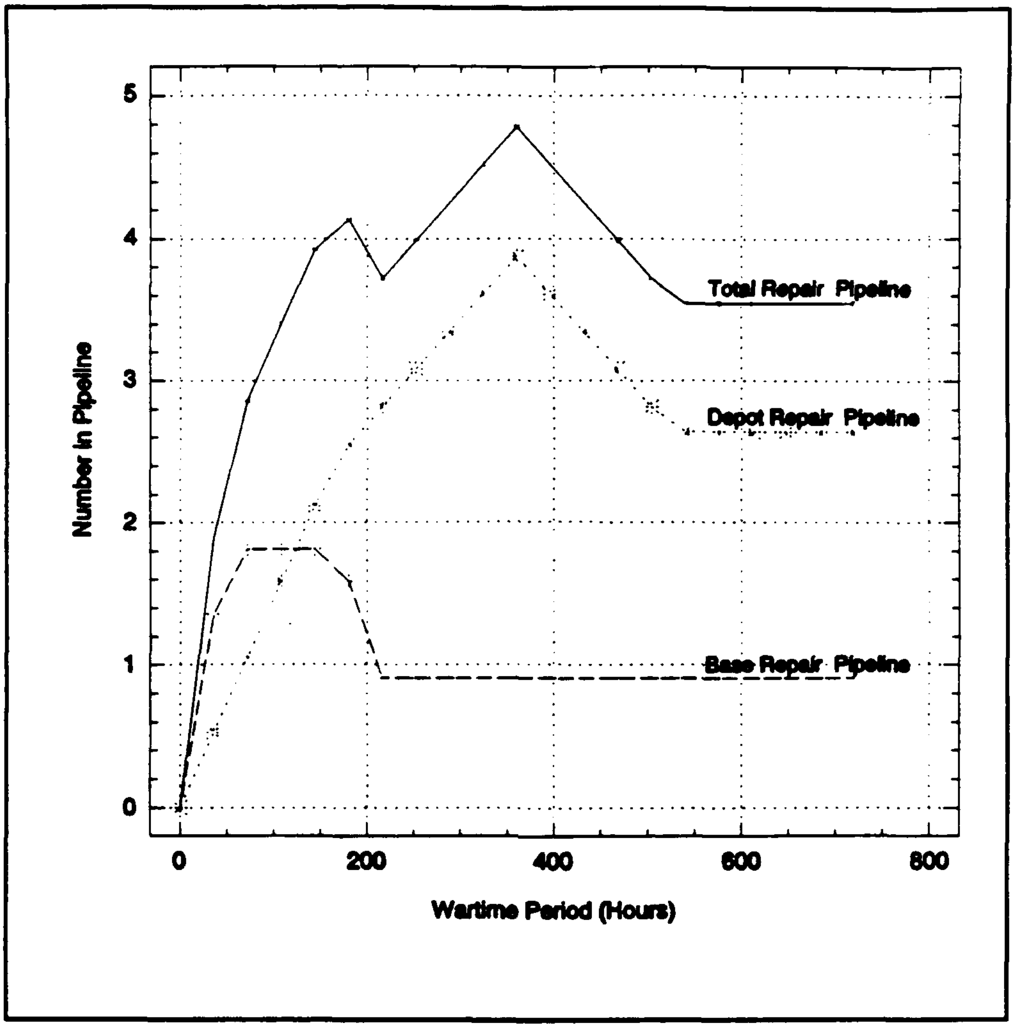

FIGURE 3-1: NONHOMOGENEOUS POISSON PIPELINES.

(2) Nonhomogeneous Compound Poisson Pipelines

In the case of nonhomogeneous Compound Poisson pipelines where the *Base Repair Pipeline* and the *Depot Repair Pipeline* have negative binomial distributions with the same *VMR*, then the resulting *Total Repair Pipeline* also has a negative binomial distribution with parameters { a, $R(t)$ } where

$$\left.\begin{aligned} a &= 1 / VMR \\ R(t) &= -\,[\Lambda^b(t) + \Lambda^d(t)] \,/\, \log(a) \end{aligned}\right\} \qquad (3.26)$$

Sherbrooke [Ref. 14] also obtained similar results for demands with geometrical compounding distributions. However, if the *Base Repair Pipeline* has a different *VMR* from the *Depot Repair Pipeline* even if both have logarithmic Poisson distributions, then the total quantity would no longer be negative binomial (i.e., the expression given in Equation 3.26 no longer applies and there is no closed form expression).

b. Total Pipeline

It is evident that more systems will be become unavailable if there are more *LRUs* being repaired at the repair facilities or in transit between these facilities. Therefore, the total expected number of failed units for an *LRU* still in repair *(Total Pipeline)* has to be ascertained in order to compute the *MMOEs* of the WSM.

In a real situation, the *Total Pipeline* is a complex convolution of both the *Base Repair Pipeline, Depot Repair Pipeline* and other logistics *Pipelines* (usually transportation). Because of this convolution of many pipelines, the outcome for the *Total Pipeline* does not have an exact closed-form expression. This subsection derives the expression for the *Total Pipeline* when all the *Pipelines* are assumed to have nonhomogeneous Poisson properties. However, the same approach is equally

applicable to *Pipelines* with nonhomogeneous compound Poisson properties under specific conditions explained earlier.

The following time-dependent notations will be used for an arbitrary *LRU*, and, to simplify these notations, the subscript i used for *LRU* type will not be included.

$\Lambda^b{}_k(t)$ = *Base Repair Pipeline* generated from base k;

$\Lambda^d{}_k(t)$ = *Depot Repair Pipeline* generated from base k;

$\Lambda^f{}_k(t)$ = *Forward Pipeline* generated from base k, in transit from base k to depot;

$\Lambda^r{}_k(t)$ = *Return Pipeline* generated from base k, in transit from depot k to base;

$S^d(t)$ = Supply level at the depot;

$S_k(t)$ = Supply level at the base k;

$T^f{}_k$ = Transportation time from base k to depot (Forward Time);

$T^r{}_k$ = Transportation time from depot to base k (Return Time).

Only one repair base and one repair depot will be considered. Figure 3-2 illustrates the case. The transportation *Pipelines* are included in the analysis.

When no inventory of the *LRU* is held at the depot, a defective *LRU* sent to the depot must be repaired at the depot before it can be returned to the base for stocking. For convenience, let $\Lambda^{fd}{}_k(t) = \Lambda^f{}_k(t) + \Lambda^d{}_k(t)$ and refer to this term as the *Depot Forward Pipeline*.

In this case, the expected *Total Pipeline* is given by

$$\Lambda(t) = \quad \Lambda^{b}_{k}(t) + \Lambda^{fd}_{k}(t) + \Lambda^{r}_{k}(t) \ . \tag{3.27}$$

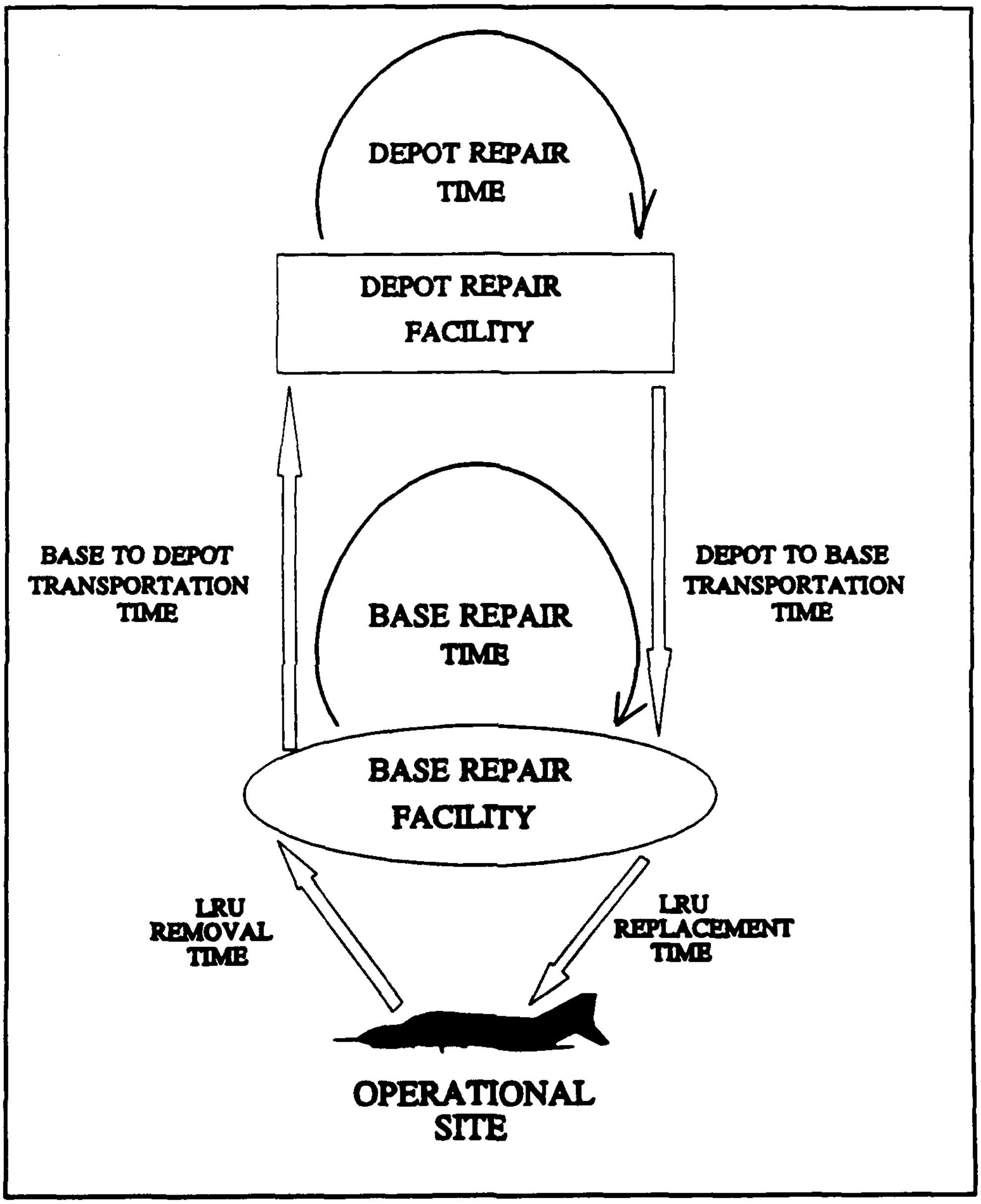

FIGURE 3-2 : A SIMPLE TWO-ECHELON MAINTENANCE SYSTEM.

On the other hand, when inventory is available at the depot, the base can now depend on the depot inventory to reduce the waiting time for *LRUs* that were sent to the depot. Instead of $\Lambda^{fd}{}_k(t)$, the base will only feel the shortages due to the unfilled orders placed at the depot. Let this expected number of *Backorders* at time t be $EB^d[\, S^d(t),\ \Lambda^{fd}{}_k(t)\ ]$. Then the depot portion of the *Total Pipeline* at base k, denoted by $\Lambda^u{}_k(t)$, is the sum of these *Backorders* and the *Return Pipeline.* Thus,

$$\Lambda^u{}_k(t) = \quad EB^d[\, S^d(t),\ \Lambda^{fd}{}_k(t)\ ] + \ \Lambda^r{}_k(t)\ . \tag{3.28}$$

Therefore, the expected *Total Pipeline* at base k is

$$\Lambda(t) = \quad \Lambda^b{}_k(t) + \Lambda^u{}_k(t)\ . \tag{3.29}$$

However, time dependency does present difficulty in determining the $\Lambda^u{}_k(t)$ since this quantity is conditioned on the stock available at the depot, completion of the depot repair and $T^r{}_k$. A better way of looking at this quantity is to study the quantity of demands during one $T^r{}_k$ since all these will be unfilled (i.e., the shortages that exist started at time $[t - T^r{}_k]$. Thus

$$\Lambda_k^u(t) = \int_{t-T_k^r}^{t} D_k^d(s)\, ds + EB^d[S^d(t - T_k^r),\ \Lambda_k^{fd}(t - T_k^r)]\ . \tag{3.30}$$

where

$D^d{}_k(s)$ = The depot failure arrival rate of a particular *LRU* generated from base k starting at time *s*.

For multiple bases served by only one depot, the approach is the same except that the depot supply level, $S^d(t)$ now is used by all the bases. Therefore

$$\Lambda^{fd}(t) = \sum_{k=1}^{K} \Lambda_k^{fd}(t) \quad , \tag{3.31}$$

and the expected number of shortages in the *Depot Forward Pipeline* is given by $EB^d[S^d(t), \Lambda^{fd}(t)]$.

The next step is to allocate these shortages to all the bases depending on the choice of the repair policy. One alternative suggested by Sherbrooke [Ref. 15] for steady-state cases is to allocate based on the relative quantity of demands placed at time t. That is,

$$EB_k^d(t) = \frac{D_k^d(t)}{\sum_{k=1}^{K} D_k^d(t)} EB^d[S^d(t), \Lambda^{fd}(t)] \ . \tag{3.32}$$

However, sudden changes in demand at the bases may render this technique rather unstable. A more stable method suggested by Hillestad [Ref. 16] is to employ time-averaged demands at the bases. To do this, let $DA^d_k(t,\delta)$ be the average demands in interval $[t-\delta,t]$ where δ is a small interval of time. Therefore

$$DA_k^d(t,\delta) = \int_{t-\delta}^{t} D_k^d(t)\, dt \ . \tag{3.33}$$

Hence the new allocation rule is:

$$EB_k^d(t) = \frac{DA_k^d(t,\delta)}{\sum_{k=1}^{K} DA_k^d(t,\delta)} EB^d[S^d(t), \Lambda^{fd}(t)] \ . \tag{3.34}$$

The influence of past demands can be fine-tuned by changing the value of δ. Reference 16 also observed that a larger δ value would increase the influence of past demands and a good value would be near the average time the depot takes to reallocate the shortages. Therefore using $\delta = t$ will keep the allocated number of *Backorders* below the expected *Total Pipeline* from time *{0 to t}*. The approach is also widely used by Sherbrooke [Ref. 15].

F. THE SMOEs

It has been shown in section E of this chapter that the expected *Total Pipeline* experienced by a base is dependent on the various pipelines of the logistics flow depicted in Figure 3-2 and the available inventory levels at the base and the depot. Combining these elements allows us to assess various measures describing the availability of *LRUs* required to keep the systems operationally ready.

Although these *SMOEs* do not directly describe the system mission readiness, they do form a basis for computing *MMOEs* and also are useful in identifying *LRUs* that cause the system to be unavailable for missions. This subsection describes the analytical approach to derive these *SMOEs*.

1. $EB_i(t)$, Expected Number of Backorders for LRU_i

The computation of the expected number of backorders (also referred to as *"Backorders"* in this thesis) for a particular LRU_i, denoted by $EB_i(t)$, depends on whether the pipelines follow the nonhomogeneous Poisson process or the nonhomogeneous Compound Poisson process.

a. *$EB_i(t)$ for Nonhomogeneous Poisson Pipelines*

Suppose the initial inventory level for a particular LRU_i is $S_i(t)$. The purpose of $S_i(t)$ is to protect the systems against the *Total Pipeline* demands for spares of the *LRU*. When that demands exceeds $S_i(t)$, the LRU_i is said to be in a "backordered" state. Since systems require this *LRU* to be *Mission-Capable,* any *Backorder* situation affects the *MMOEs.*

Suppose that the expected *Total Pipeline* $\Lambda_i(t)$ for LRU_i follows the nonhomogeneous Poisson distribution, the $EB_i(t)$ at time t is given by:

$$\begin{aligned} EB_i(t) &= \sum_{k=S_i(t)+1}^{\infty} [k - S_i(t)]\, P[k\,;\, \Lambda_i(t)] \\ &= [\Lambda_i(t) - S_i(t)] + \sum_{k=0}^{S_i(t)} [S_i(t) - k]\, P[k\,;\, \Lambda_i(t)]\,. \end{aligned} \tag{3.35}$$

The latter expression is easier to program on a computer.

To complete the analysis, the variance of the *Backorders* is given by

$$VB_i(t) = \sum_{k=S_i(t)+1}^{\infty} [k - S_i(t)]^2 \, P[k \, ; \, \Lambda_i(t)] - [EB_i(t)]^2$$

$$= \Lambda_i(t) + [\Lambda_i(t) - S_i(t)]^2 - [EB_i(t)]^2 - \sum_{k=0}^{S_i(t)} [k - S_i(t)]^2 \, P[k \, ; \, \Lambda_i(t)] \, . \qquad (3.36)$$

Again, the latter expression is easier to program on a computer.

b. $EB_i(t)$ for Nonhomogeneous Compound Poisson Pipelines

The exact expression of $EB_i(t)$ for nonhomogeneous Compound Poisson pipelines can be carried out in a similar manner as shown in the previous paragraph and has been done by Sherbrooke [Ref. 15].

2. EB(t), the Total Backorders of the Deployed Systems

The *Total Backorders*, denoted *EB(t)*, is an *SMOE* which measures the expected total number of all *Backorders* experienced by all the deployed systems of the military capability. If there are *I* types of *LRUs* then *EB(t)* at any time *t* can be computed as

$$EB(t) = \sum_{i=1}^{I} EB_i(t) \quad . \qquad (3.37)$$

With a policy of no cannibalization, *EB(t)* is a parameter used in the computation of the most conservative value for *Ao(t)*. The exact expression for *Ao(t)* is shown below in Equation 3.41.

3. Operational Availability (Ao(t))

Suppose that there are a total of *I* types of *LRUs* for each system and the total number of deployed systems are assumed to be identical both in hardware and functions. In the computation of *Ao(t)*, all of the *LRUs* are considered essential for the operation of the system and therefore their failures cause the system to fail. It is important to note here that the computation to be described later can be extended to handle different types of systems. Similar to the OPUS-8 algorithm, the overall *Ao(t)* for the whole deployment can be obtained by taking the weighted average of all of the individual systems' *Ao(t)*.

a. *Ao(t) without Cannibalization*

Here the policy of not consolidating shortages of *LRUs* among the systems is adopted. The probability that a single military capability consisting of a total number of *ns(t)* systems at time *t*, is short of LRU_i, given that there are *b(t)* *Backorders* of LRU_i at an arbitrary base at the same time , can be written as *B(t)/ns(t)* under the same conditions used to derive Equation 3.21. Assume for now that there is only one of each type of *LRU* in each system, then the probability that these *ns(t)* systems has a shortage of LRU_i is computed by summing over all possible $b(t) = k - S_i(t)$ values multiplied by the probability that they occur, which is

$$\sum_{k=S_i(t)+1}^{\infty} \frac{[k - S_i(t)]}{ns(t)} P[k \; ; \Lambda_i(t)] = \frac{EB_i(t)}{ns(t)} \quad . \tag{3.38}$$

Assuming that the *LRU* failures are independent and noting that the probability of no shortage of each is the complement of the probability given by Equation 3.38, then using the reliability theory of items-in-series, the probability that the whole deployment is *Mission-Capable* is given by

$$Ao(t) = \prod_{i=1}^{I} [\ 1 - \frac{EB_i(t)}{ns(t)}]\ . \tag{3.39}$$

As mentioned in the previous subsection, there is a relationship between *Ao(t)* and *EB(t)* which can be shown by expanding Equation 3.39 as follows.

$$Ao(t) = [\ 1 - (\sum_{i=1}^{I} \frac{EB_i(t)}{ns(t)} + \sum_{i=1}^{I-1} \sum_{j=i+1}^{I} \frac{EB_i(t)}{ns(t)} \cdot \frac{EB_j(t)}{ns(t)} + \ldots)]\ , \tag{3.40}$$

where

$EB_j(t)$ = Backorders of LRU_j at time *t*.

When individual $EB_i(t)$ are very small as compared to $ns(t)$, then the second and higher terms of Equation 3.40 become negligible. Therefore, Equation 3.40 can be reduced to the approximate expression

$$Ao(t) \approx [\ 1 - \frac{EB(t)}{ns(t)}]\ . \tag{3.41}$$

If there are several units of LRU_i, denoted by q_i, in each system, Hillestad [Ref.16] showed that Equation 3.39 can to be modified to

$$Ao(t) = \prod_{i=1}^{I} \sum_{y=0}^{ns(t)q_i} \frac{\binom{ns(t)q_i - 1}{q_i}}{\binom{ns(t)q_i}{q_i}} PB_i(y) \,, \tag{3.42}$$

where $PB_i(y)$ is the probability that LRU_i has y shortages at time t with the following conditions

$$PB_i(y) = \begin{cases} \sum_{k=0}^{S_i(t)} P[k \; ; \; \Lambda(t)] & \text{if } y = 0 \\ \\ P[y + S_i(t) \; ; \; \Lambda(t)] & \text{if } y > 0 \,. \end{cases}$$

b. Ao(t) with Cannibalization

Again, suppose that there are I types of *LRUs in* each system and *ns(t)* is the number of identical systems. Under the cannibalization policy, shortages of *LRUs* can be consolidated into the smallest possible number of systems. Cannibalization is often practiced in military establishments to minimize the number of Not-Mission-Capable Systems *(NMCS)* in time of war. The rationale for doing this is that when a system becomes unavailable, it does not matter if there is only one missing *LRU* or more. Therefore, the good *LRUs* in a failed system can be used as spares to keep other systems *Mission-Capable.*

Suppose that there is only one of each type of *LRU* in the system and let $P^i(j)$ be the probability that the number of shortages of the LRU_i are less than or

equal to j. Also, assuming that the $\Lambda_i(t)$, the *Total Pipeline* of the LRU_i, is from a nonhomogeneous Poisson process, then

$$P^i(j) = \sum_{k=0}^{S_i(t)+j} P(k\ ;\ \Lambda_i(t)) \ . \tag{3.43}$$

Now let $P(j)$ be the probability that the number of *NMCS* is less than or equal to j. Here j is the maximum number of *NMCS* which can be tolerated by the users. With the cannibalization policy, we can concentrate all the shortages of *LRUs* to the minimum number of *NMCS* (i.e., there are j *NMCS* if all its *LRUs* also have j shortages). Assuming that each system is made up of I types of *LRUs* connected in series, then $P(j)$ can be obtained by the multiplication of all its *LRUs'* $P^i(j)$. When there is only one of each *LRU* in each system, $P(j)$ is

$$P(j) = \prod_{i=1}^{I} P^i(j) \ . \tag{3.44}$$

In the case where there are q_i units of LRU_i in each system, these units can be used as spares for cannibalization if the system becomes *NMC*, then

$$P(j) = \prod_{i=1}^{I} P^i(q_i\, j) \ . \tag{3.45}$$

From the theory of stochastic processes, the expected value of a nonnegative discrete distribution can be computed from the sum of the values of the complementary *CDF* of that distribution. Since the upper limit of the summation is the number of

systems, $ns(t)$, being deployed at time t, then the expected number of *NMCS* at time t, $ENMCS_c(t)$, is

$$ENMCS_c(t) = \sum_{j=0}^{ns(t)} [1 - P(j)] \ . \tag{3.46}$$

Then $Ao(t)$ with Cannibalization, denoted by $Ao_c(t)$, is

$$Ao_c(t) = \frac{[ns(t) - ENMCS_c(t)]}{ns(t)} \ . \tag{3.47}$$

G. The MMOEs

The *SMOE*, $Ao(t)$, computed in the previous subsection can also be used to derive the *MMOEs* which are measures more readily understood by the operators of the system. Although *ENMCS* is of main interest, the formulation for $P[ENMCS \leq NMCS]$ is also given here. The definitions of both *MMOEs* were given in section II-C1.

1. ENMCS

Without Cannibalization, the expected number of *NMCS* at time t is

$$ENMCS(t) = ns(t) - [ns(t)\ Ao(t)], \tag{3.48}$$

Depending on whether there is one or more of each *LRU*, the expression of Ao(t) can be obtained either from Equations 3.39 or 3.42.

In the case of Cannibalization, the expression for *ENMCS(t)* is exactly the same as Equation 3.46.

2. P(ENMCS ≤ NMCS)

The analyses described in subsection III-F3 are useful in the formulation of *P(ENMCS ≤ NMCS)* with or without cannibalization. There *j* represents the maximum number of *NMCS*. Therefore, Equations 3.44 or 3.45 can also be used as *P(ENMCS ≤ NMCS)*. Since this *MMOE* is also time dependent, there is a value of P(ENMCS ≤ *NMCS)* for each time interval of the wartime period.

H. LIMITATIONS OF THE ANALYTICAL APPROACH

The most obvious limitation of the analytical model for the WSM is the need to make certain simplifying assumptions so that closed-form expressions can be obtained.

However, in a realistic wartime period the demands for spares are intense and highly unpredictable, and there will always be queuing at repair facilities due to the limited availability of spares and repair resources. Also, in the course of deployment management will try to take actions which it thinks may optimize the *MOEs*.

Therefore, a Monte Carlo simulation model is needed to analyze the effects of more realistic scenarios and to verify whether the actions taken intuitively by management actually improve the ability of the logistics support system to meet the

operational demands. The simulation approach for the WSM is developed in Chapter IV.

IV. A SIMULATION VERSION OF THE WSM

A. REASON FOR THE SIMULATION APPROACH

As stated in section H of Chapter III, the limitations of the analytical approach for the WSM warrants the need to perform Monte Carlo sampling to simulate more realistic scenarios so that the effects on the *MOEs* of the WSM can be studied. This chapter describes the design and the implementation of a simulation version of the WSM.

B. MODEL DESIGN

The simulation model design for the WSM has a structure similar to the analytical approach as shown in Figure 3-2. The design takes the form of a multi-server, multi-job-class queuing simulation with time-varying demand (arrival) rates for each *LRU* of the system. Unlike most queuing simulations which are designed to study the steady-state behavior of a system with constant parameters, the WSM simulation studies the behavior of a group of systems deployed as one military capability and its logistics support over a finite time period during which the demand rates vary dynamically.

The flow of the various processes of the model are discussed in the following sections. Issues concerning the need to replicate runs and their statistical effects on the *SMOEs* and *MMOEs*, sampling interarrival times, handling of random number

streams, repair selection priority rules, and base and depot repair policies are also addressed. The main processes to be modelled for the WSM Simulation are provided in Table 4.1.

TABLE 4.1 : DESIGN OF FLOW PROCESSES FOR THE SIMULATION MODEL

STEP	PROCESS	PROCESS CHARACTERISTICS
1.	*LRU* Failure Arrivals	Computation based on input parameters.
2.	Failure Removal	A process with unlimited resources and random or fixed parameters.
3.	Failure Rectification	A process constrained by number of available spares and cannibalization polices.
4.	Base Repair	A process with limited or unlimited resources and random or fixed parameters.
5.	Base-to-Depot Transportation	A process with unlimited resources and fixed parameters.
6.	Depot Repair	A process with unlimited resources and random or fixed parameters.
7.	Number of Demands	Simulation output (combination of Base and Depot demands).
8.	Number of Backorders - $EB_i(t)$	Simulation output. Depends mainly on processes 3 and 7.
9.	Number of Mission-Capable Systems	Simulation output. Depends mainly on processes 3 and 8.
10.	*Ao(t) - (SMOE)*	Computation based on process 9.
11.	*ENMCS(t) -(MMOE)*	Computation based on process 9.

1. LRU Failure Arrivals

Each *LRU* failure in a system causes the system to be Not-Mission-Capable *(NMC)*. A time-dependent variable, to be defined as the Mean Time

Between Demands *(MTBD)* for each *LRU,* is computed based on the inverse of the demand rate, *D(t),* which has a formula given by Equation 3.1. *MTBD* is not a constant parameter since the value of *D(t)* changes according to the status of the utilization rate and the number of Mission-Capable Systems *(MCS)* at a particular time interval of the wartime period. To generate a given interarrival time for each *LRU* of the entire deployment of systems, we sample from an exponential random number stream using *MTBD* as the mean parameter.

2. Failure Removals

When the system becomes Not-Mission-Capable *(NMC),* time is spent on isolating the cause of the failure and the removal of the faulty *LRU.* An exponential time or a constant time is used to generate this isolation and removal time.

3. Failure Rectification

An *NMC* system can be recovered by replacing the faulty *LRU* with a good spare. However, the *NMC* system remains in the same status if there is a shortage of spares for that *LRU.*

When the policy of no cannibalization is adopted, the model treats the failure of an *LRU* as a system failure (i.e, one system becomes *NMC*). If there is an available spare for that failed *LRU* in the inventory of stock, the *NMC* system will be restored to a Mission-Capable *(MC)* status. When no more spares are left, the failed system must then wait for a repaired *LRU* of the same type to be returned to the inventory storage facility. Repair prioritization polices that were discussed in

subsection II-B5 were used to determine which *NMC* system is entitled to the newly arrived repaired *LRU.* Additional provisioning of spares to satisfy unfilled *Backorders* is not considered in this simulation in view of the fact that WSM is presently only an assessment tool.

The cannibalization policy is more complex to model. As long as there are spares in the inventory, no cannibalization of other *NMC* systems will be carried out to restore a system that just had a failure. In this case, that system is considered *MC* after a fixed time needed for replacing the faulty *LRU* with a spare. When there is no spare left in the inventory, cannibalization will start with the first *NMC* system. A search for the required *LRU* will be made from the *NMC* systems until a good *LRU* can be found. Then the system that became *NMC* last can be repaired using a good *LRU*. If no *LRU* is found, then the system will remain *NMC* until a repaired *LRU* is returned to the stock. A limit can be placed on the maximum number of systems that can be cannibalized if such a requirement is stated in the WMP.

4. Repair Flows

Similar to the analytical approach, a failed *LRU* can either be repaired at the base or sent to the depot for repair depending on its Not-Repairable-This-Station *(NRTS)* value (see Equation 3.24). A uniform(0,1) random stream is used to generate the random variable. If the random variable is less than the *NRTS* value,

then the failed *LRU* is sent to the depot for repair. Otherwise, it goes to the base for repair.

a. Base Repair

If the faulty *LRU* is to be repaired at the base, the model allows the specification of a number of base repair stations in order to study the effect of limited base repair on the *SMOEs* and *MMOEs*. In the example of a one-base, one-depot maintenance system to be studied later in Chapter VI, these repair stations are all located at one base and all failed *LRUs* will form a single queue to be served by these stations. Different policies to prioritize repair are also allowed. The base repair facility will service a failed *LRU,* either with an exponential time or a constant time, using an input parameter which is specified for that *LRU.*

b. Depot Repair

Repair at the depot requires three time components - fixed transportation time from a base to a depot, a fixed or an exponential depot repair time and a fixed transportation time from a depot to a base.

5. Replications for Statistical Significance

A single replication of the model will cover the entire wartime period. Many replications of the simulation must be conducted to achieve satisfactory statistical confidence intervals for the outputs. To measure the results as a function of time, each replication is further divided into smaller equal time intervals.

6. Random Number Streams and Generation

All of the random processes indicated in Table 4.1 require Monte Carlo sampling. To minimize the number of random number streams, all the *LRUs* will use the same stream for their arrival generations. The other processes will each have a dedicated random number stream.

7. Repair Prioritization Policies

An appropriate choice of repair prioritization is important when repair resources are limited since a decision to repair one *LRU* also means a decision not to repair another *LRU* when both are competing for the same repair resources. Two widely known policies will be considered.

a. First Come First Served (FCFS)

The *FCFS* priority selects the *LRU* which has been waiting the longest for repair. Therefore, this prioritization policy does not consider those *LRUs* that may have the greatest demands for spares or repairs.

b. Least Available Item First (LAIF)

The Least Available Item First *(LAIF)* repair priority looks for the *LRU* with the highest $EB_i(t)$ to service first since filling a backorder will improve the $Ao(t)$.

C. IMPLEMENTATION OF WSM USING MODSIM

MODSIM II is a fourth generation programming language recently acquired by the Naval Postgraduate School for developing complex simulation models. Its features, programming syntax, and structure are described in the MODSIM Manuals [Ref. 4]. MODSIM was chosen to develop the WSM Simulation in view of its powerful features to handle an object-oriented discrete event simulation. The WSM Simulation is implemented using a PC-based version of MODSIM II. Hence the simulation program was limited by the PC base memory of 640K and this was one main reason for not being able to incorporate aspects such as condemnation of *LRUs* and an intermediate repair level into the simulation.

Figure 4-1 depicts the main design flow and linkage between the various modules of the WSM simulation program as used in this thesis.

The program codes and listings are given in Appendix A.

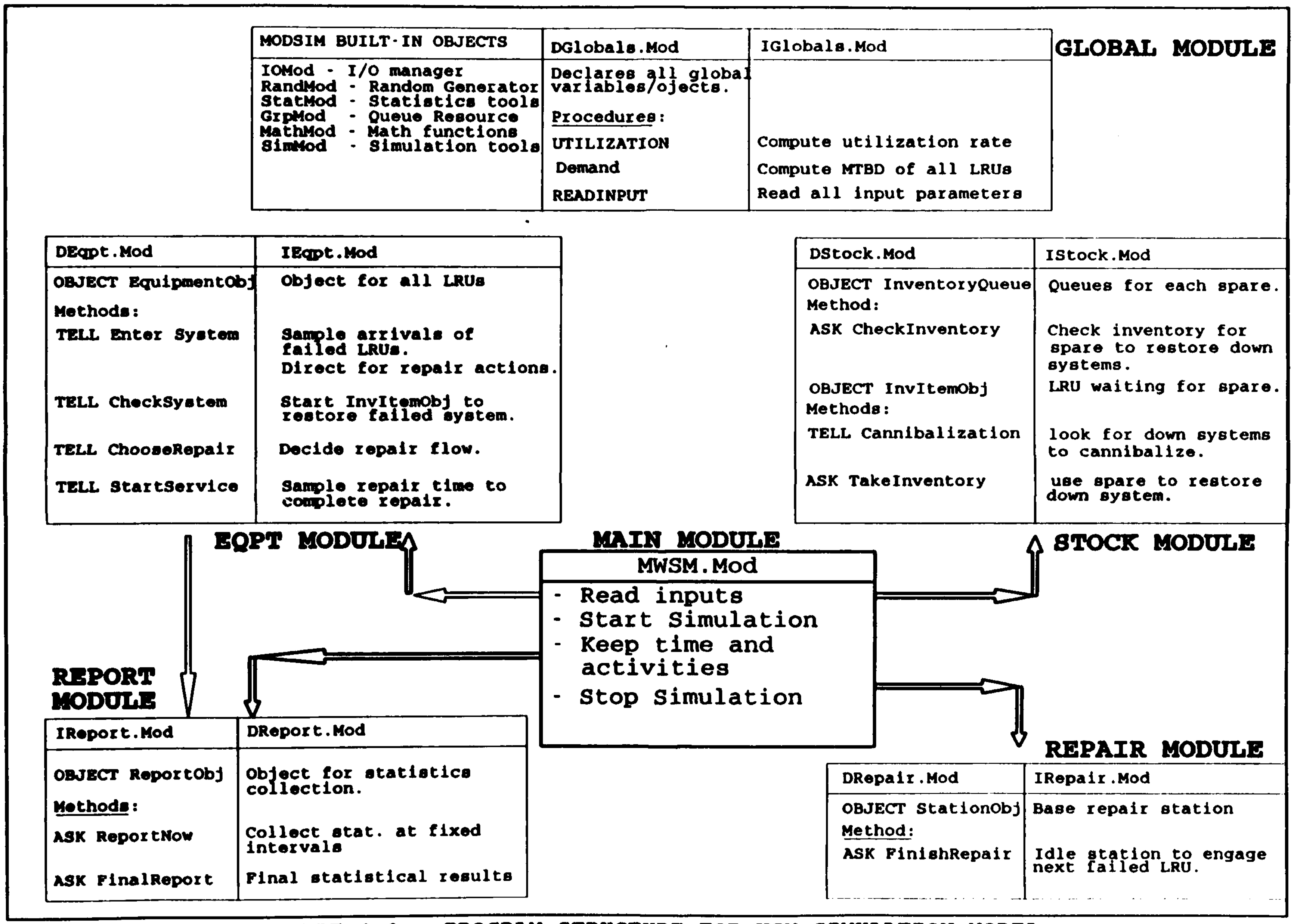

FIGURE 4.1: PROGRAM STRUCTURE FOR WSM SIMULATION MODEL

D. STATISTICAL INFERENCES

This section describes the statistical analyses that were performed on the inputs and the outputs of the simulation runs to verify that these two elements were statistically sound. STATGRAPHICS [Ref. 17] was used to enhance the analyses.

1. The Resulting Distribution for Failure Arrivals

The arrival of failures for each *LRU* is generated from an exponential random stream using *MTBD* as the parameter. Since *MTBD* is not a constant, the distribution for the interarrival times of failures for each *LRU* is not expected to belong to any one single family of commonly known distributions such as exponential and gamma. To verify this, data for the interarrival times of each *LRU* were collected for statistical analyses to assess whether they could be fitted to any of the commonly known distributions.

An exponential distribution was first used to attempt to fit 3001 data points obtained for the interarrival times of *LRU D* of the example in chapter VI and, at first glance, Figure 4-2 suggests that the fit seems to be good. However, on further examination using the Kolmogorov-Smirov *(K-S)* goodness of fit test, the exponential distribution was rejected based on the test results given in Table 4.2. A Chi-square goodness of fit test was also performed with the same conclusion. The results are shown in Table 4-3. Further attempts to fit the data with other distributions such as Gamma, Normal and Lognormal were also unsuccessful.

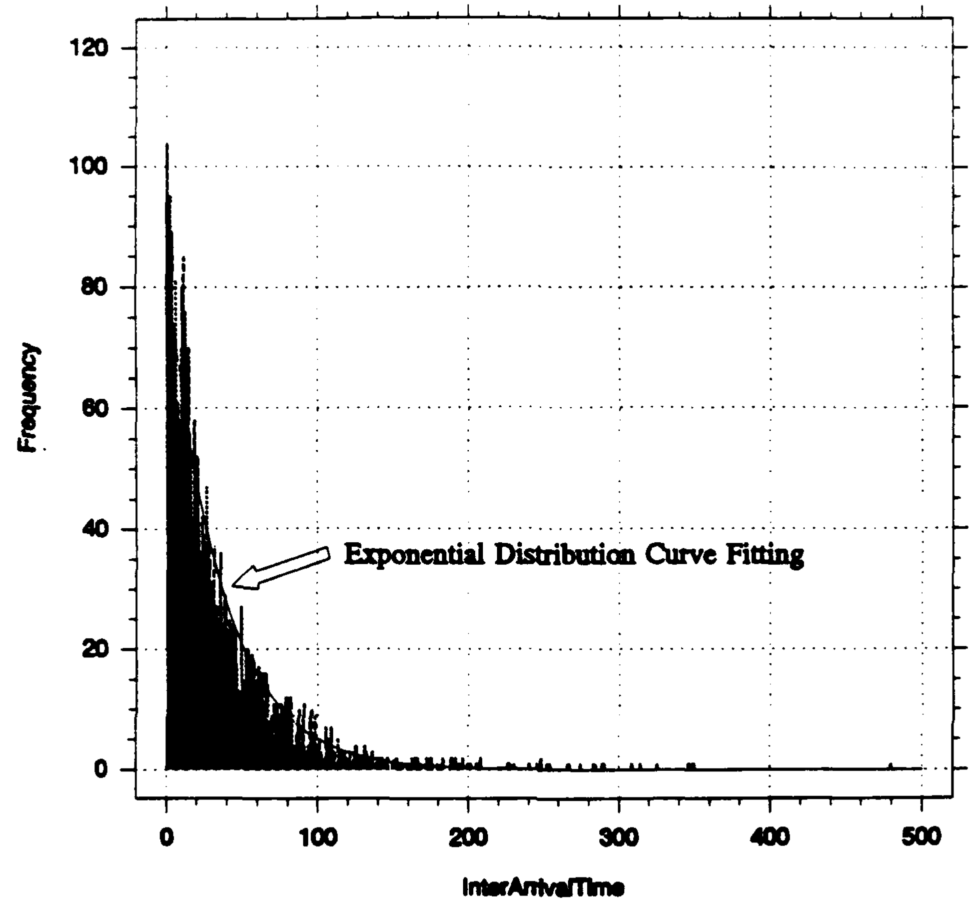

FIGURE 4-2: EXPLORATORY ANALYSIS USING EXPONENTIAL DISTRIBUTION FITTING ON 3001 DATA POINTS OBSERVED FOR THE INTER-ARRIVAL TIMES OF LRU D.

TABLE 4.2 : K-S TEST ON INTERARRIVAL TIME DISTRIBUTION		
Estimated KOLMOGOROV statistic DPLUS	=	0.0434898
Estimated KOLMOGOROV statistic DMINUS	=	0.0107071
Estimated overall statistic DN	=	0.0434898
Approximate significance level (Therefore not a good fit)	=	2.3493E-5

TABLE 4-3 : CHI-SQUARE TEST ON INTERARRIVAL TIME DISTRIBUTION

Lower Limit	Upper Limit	Observed Frequency	Expected Frequency	Chisquare
at or below	4.286	377	338.5	4.37296
4.286	21.429	1099	1012.9	7.31331
21.429	38.571	563	627.6	6.64243
38.571	55.714	351	388.8	3.67673
55.714	72.857	218	240.9	2.17470
72.857	90.000	137	149.2	1.00432
90.000	107.143	88	92.5	.21550
107.143	124.286	48	57.3	1.50532
124.286	141.429	36	35.5	.00728
141.429	158.571	17	22.0	1.13197
158.571	175.714	19	13.6	2.12192
175.714	192.857	14	8.4	3.66202
192.857	210.000	11	5.2	6.36820
above	210.000	23	8.5	24.63549

Chisquare = 64.8321 with 12 d.f. Sig. level = 2.92816E-9
(Therefore not a good fit)

All the above are indications that one cannot expect to obtain exact analytical expressions for the *SMOEs* and the *MMOEs* when the distribution of the failure arrivals of each *LRU* does not belong to a single family of known distributions.

2. Replications for Statistical Confidence

In any simulation, many replications are required so that a reasonable statistical confidence can be obtained for all the outputs. In the case of the WSM simulation, each replication "opens" at time $t=0$ under the same set of initial conditions and then "closes" at the end of the same wartime period, which is fixed. Under these conditions, the method of independent replications (see Banks and Carson [Ref. 18], p.421-422) is used for this type of terminating simulation. The whole simulation was repeated many times, with each replication following the conditions described above and using a different random number stream. Therefore, all the data points from the replications collected at a particular time interval for a particular output are statistically independent and identically distributed. Banks and Carson concluded that the classical methods of confidence interval estimation using the *Student t* distribution can be applied to the output. In this simulation, a 95% confidence interval was used as the stopping criterion for the whole simulation run.

V. WSM BY AN OPUS-8 APPROXIMATION

Due to the unavailability of a proper analytical tool to assess the sustainability of a military capability under wartime environment, the MINDEF logistics staff is currently using OPUS-8 to roughly estimate the requirements of war-reserve spares. Despite the fact that OPUS-8 is a steady-state model, it remains a popular evaluation and optimization tool in MINDEF for assessing the adequacy of spares to support a military capability. In the past, the author was personally involved in trying to adapt OPUS-8 as a rudimentary sustainability tool but the approach had not been successful due to the lack of understanding of the dynamic behavior of a military asset under a wartime environment. Having gained understanding of the WSM through the development of the analytical and simulation models, the author attempts in this chapter to formulate a proper methodology to allow OPUS-8 to be used as an approximation tool for the WSM.

A. THE STEADY-STATE ASSUMPTIONS OF OPUS-8

1. Current Features and Assumptions

OPUS-8's main strength lies in its superb ability to handle complex multi-echelon logistics support and multi-indenture system structure. Figure 5-1 illustrates a multi-echelon logistics support with multi-system deployment and Figure 5-2 the multi-indenture breakdown of a system.

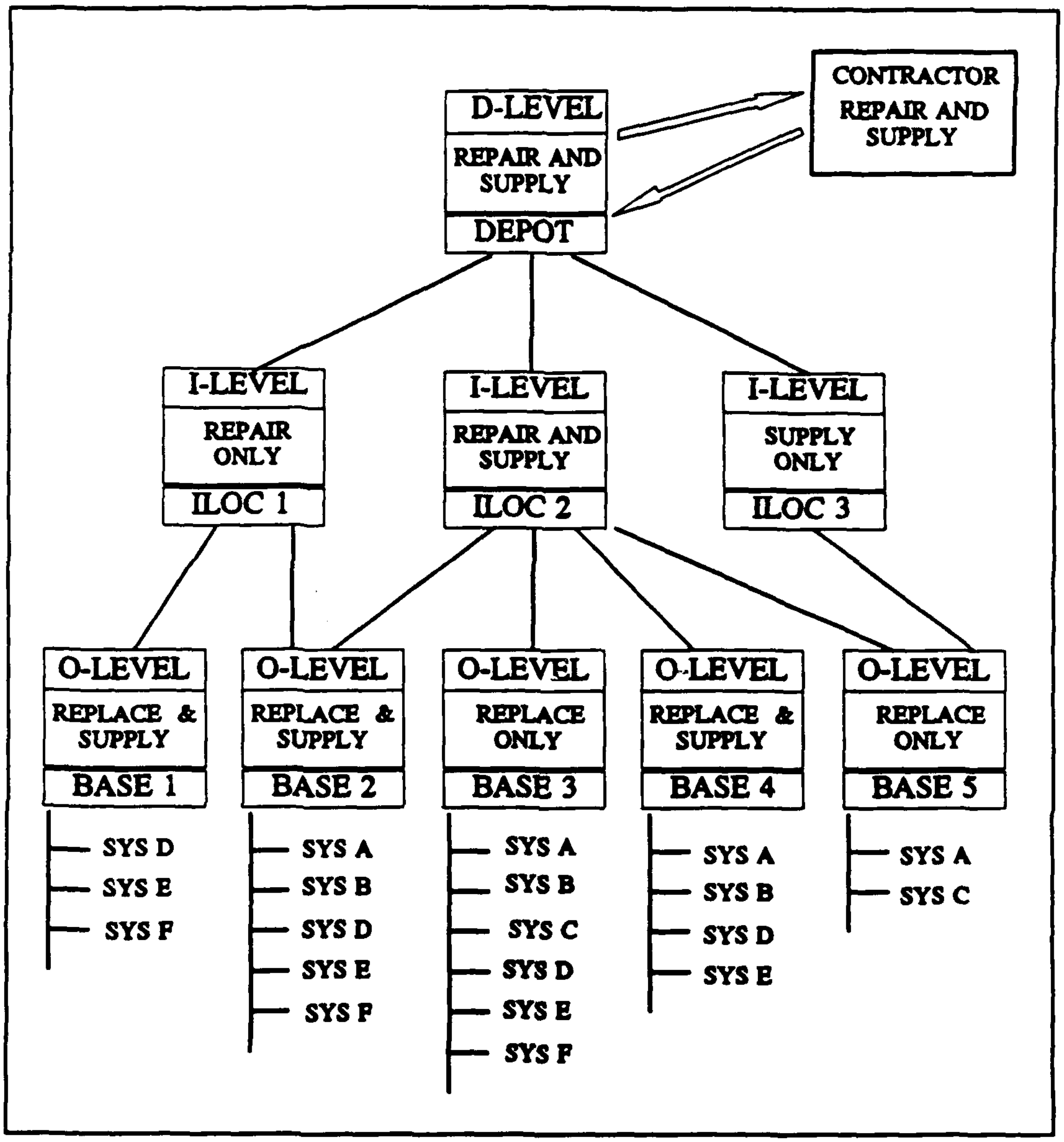

FIGURE 5-1 : MULTI-ECHELON LOGISTICS SUPPORT WITH MULTI-SYSTEM DEPLOYMENT.

LEGENDS:

O-LEVEL - Organization Level Maintenance
I-LEVEL - Intermediate Level Maintenance
D-LEVEL - Depot Level Maintenance
ILOC - Intermediate-Level Location.

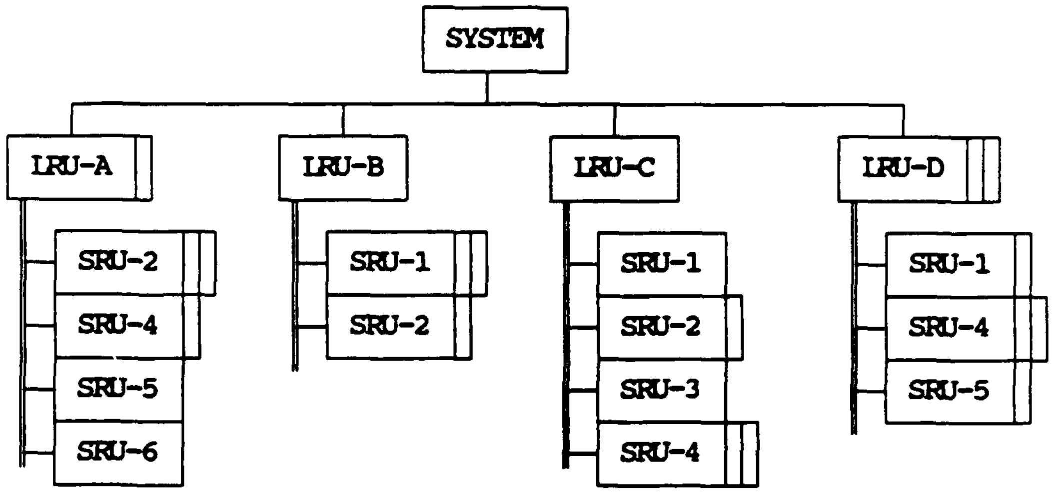

FIGURE 5-2 : MULTI-INDENTURE SYSTEM STRUCTURE

Sophisticated techniques such as network and dynamic programming are imbedded in the OPUS-8 optimization algorithms [Ref.4]. The user has a choice of different *SMOEs* (Effectiveness) to be used together with spares investments (Cost) as the optimization criteria. A single run will produce a Cost-Effectiveness curve, with each optimal point on the curve providing an investment amount corresponding to an optimized range and depth of spares. In addition to the optimization techniques, OPUS-8 uses many well-known steady-state analytical formulae from stochastic and inventory theories based on the following assumptions:

"a. the number of demands in a given time interval follows a Poisson distribution with a known constant mean value. However, at the operational level, the number of demands in a given time interval is allowed to follow a Geometric Compound distribution with a given value on the Variance-to-Mean Ratio *(VMR).* With this type of demand distribution batching of demands can be modelled.

b. The mean values of the Turn Around Times are known. (No other assumptions are made on the distributions of the Turn Around Times.)

c. A failure of one type of item is statistically independent of those that occur for any other type of item. However by introducing Removal Rate Factors for item multiple errors in a mother item can modelled.

d. The Turn Around Times for items in the repair cycle are statistically independent.

e. No queues are assumed to be formed at repair facilities, nor is any controlled bulk service permitted in any phase of the repair cycle" [Ref. 4].

2. Limitations Of Steady-state Assumptions.

Steady-state assumptions are fine for peace-time prediction of spares requirements since the fluctuations in demands when analyzed over a long-time will tend to stabilize. But in time of war when the operational profile of a military asset changes dynamically with time, these steady-state techniques are no longer valid. Also, the assumptions of no queuing and limited batching are not too suitable for sustainability analyses.

B. OPUS-8'S SUSTAINABILITY OPTION

All of the options of OPUS-8 are based on steady-state assumptions and they would require extensive modifications to achieve the capabilities of the WSM. However, the sustainability option[1] of OPUS-8 has features which are favorable for use as an approximation to the WSM without modifying any of its codes. Unfortunately, its main limitation is that no deployed or on-site repair capability is

[1]. OPUS-8 uses the term "Endurance" instead of sustainability [Ref. 4]. However the meanings of both are equivalent in this thesis.

allowed for the systems deployed as a military capability during a given the wartime period.

Similar to the WSM, *MOEs* such as the *Total Pipeline* and *Backorders* for each *LRU* and the expected number of *NMCS* are obtainable from the sustainability option of OPUS-8. Therefore, these *MOEs* are to be interpreted directly as the WSM's *MOEs* if approximation is possible.

When there is no repair, there is only one pipeline generated from the location where the systems is deployed. The pipeline will increase linearly with time if the demand rate for spares is a constant. In the case of the nonhomogeneous Poisson assumption, the pipeline increases monotonically with time during the wartime period, with its value dependent on the demand rate at the time of analysis. Therefore, at any time t the OPUS-8 option can produce an approximate result to the WSM at the same time, by using a weighted-average utilization value of all the utilization rates that have occurred over the time from zero to t. This means that multiple runs must be performed to analyze the entire anticipated wartime period. The runs are ordered so that each subsequent time period overlaps the previous time period. The rationale behind this approximation becomes evident when a numerical example is used to compare the results between the OPUS-8 approximation, the analytical model and the simulation model. These results are given in Chapter VI.

V. VERIFICATION OF THE WARTIME SUSTAINABILITY MODELS

In Chapter III, an analytical model was developed for the WSM using certain assumptions to obtain exact analytical formulae. However, when these assumptions were relaxed, the distributions of the failure arrivals for the *LRU* were found not to belong to any single family of known distributions as explained in subsection V-D1. As such, exact analytical expressions for the *SMOEs* and *MMOEs* were not attainable and therefore a simulation model was built for the WSM.

To verify that the results from the analytical model conform with the results from the simulation model, a numerical example using the similar set of input conditions was run under both models. This example was also performed with the simulation model to study the effects of wartime logistics policies such as cannibalization, limited repair resources and repair prioritization.

A special case of no repair capability during the anticipated wartime period was also investigated by both models. This case was also examined by a third approach using the sustainability option of OPUS-8 as an approximation for the WSM.

This chapter covers the comparison of the results in detail and draws conclusions about significant trends.

A. THE EXAMPLE

All the analyses mentioned in this chapter use the same numerical example. The following input parameters are used.

1. War and Mobilization Plan (WPM)

A squadron of twenty-four aircraft are deployed at only one base. The WPM in this case study anticipate a 720-hour wartime period with an initial deployment of twenty-four of the same aircraft type. Only the surge and post-tension activities were considered in this example although peacetime and pre-tension activities can also be part of the WMP. This was done to minimize the number of factors that can affect the output.

The surge period begins at time $t=0$ and lasts for 168 hours. Three sorties per day are expected from each aircraft, with each sortie having an average flight mission time of 1.6 hours. This corresponds to a daily utilization rate of 20% for each aircraft. The scenario then anticipates a daily utilization rate of 10% for the rest of the wartime period (post-tension period). An abrupt (step) change was used for the transition from the surge period to the post-tension period. This was done to simplify the analytical expressions for the analytical model. In this WMP, it is assumed the operational planners were mainly interested in estimating the *ENMCS(t)* during the anticipated wartime period.

2. System Structure and Logistic Support

Each aircraft has a one-indenture level breakdown of ten *LRUs* with input parameters as shown in Table 6-1 (not all the parameters were used each numerical example).

TABLE 6-1: INPUT PARAMETERS FOR LRUs

ITEM	MTBF	NRTS	QPA	INVLVL	BRTIME	DRTIM	REMTIM	REPTIM
A	2564.0	0.07	1	1	48.0	120.0	1.0	1.0
B	247.5	0.06	1	3	48.0	120.0	1.0	1.0
C	2777.7	0.21	1	1	72.0	120.0	1.0	1.0
D	86.9	0.59	1	5	72.0	120.0	1.0	1.0
E	222.7	0.04	1	2	48.0	120.0	1.0	1.0
F	164.7	0.04	1	4	48.0	120.0	1.0	1.0
G	1754.4	0.24	1	1	72.0	120.0	1.0	1.0
H	223.2	0.09	1	2	48.0	120.0	1.0	1.0
I	58.8	0.06	1	6	48.0	120.0	1.0	1.0
J	91.6	0.28	1	2	48.0	120.0	1.0	1.0

where

MTBF = Mean Time Between Failures in hours. This is the reciprocal of the failure rate. This characteristic is constant for each *LRU*.

NRTS = Not Repairable This Station. A value to indicate that the proportion of the repair of each *LRU* which will flow to the depot. The complement of this value is the proportion being repaired at the base. No condemnation is assumed.

QPA = Quantity Per Application. This is the quantity of each *LRU* found in each aircraft.

INVLVL = Initial Inventory Stock Level for each *LRU*.

BRTIM = Mean Repair Time (hours) at the base repair facility.

DRTIM = Mean Repair Time (hours) at the depot repair facility.

REMTIM = Mean Removal Time (hours) at the system level which includes fault isolation time and removal time. This time component was used only for the simulation model.

REPTIM = Mean Replacement Time (hours) at the system level to replace a faulty *LRU* with a good spare. This time component was used only for the simulation model.

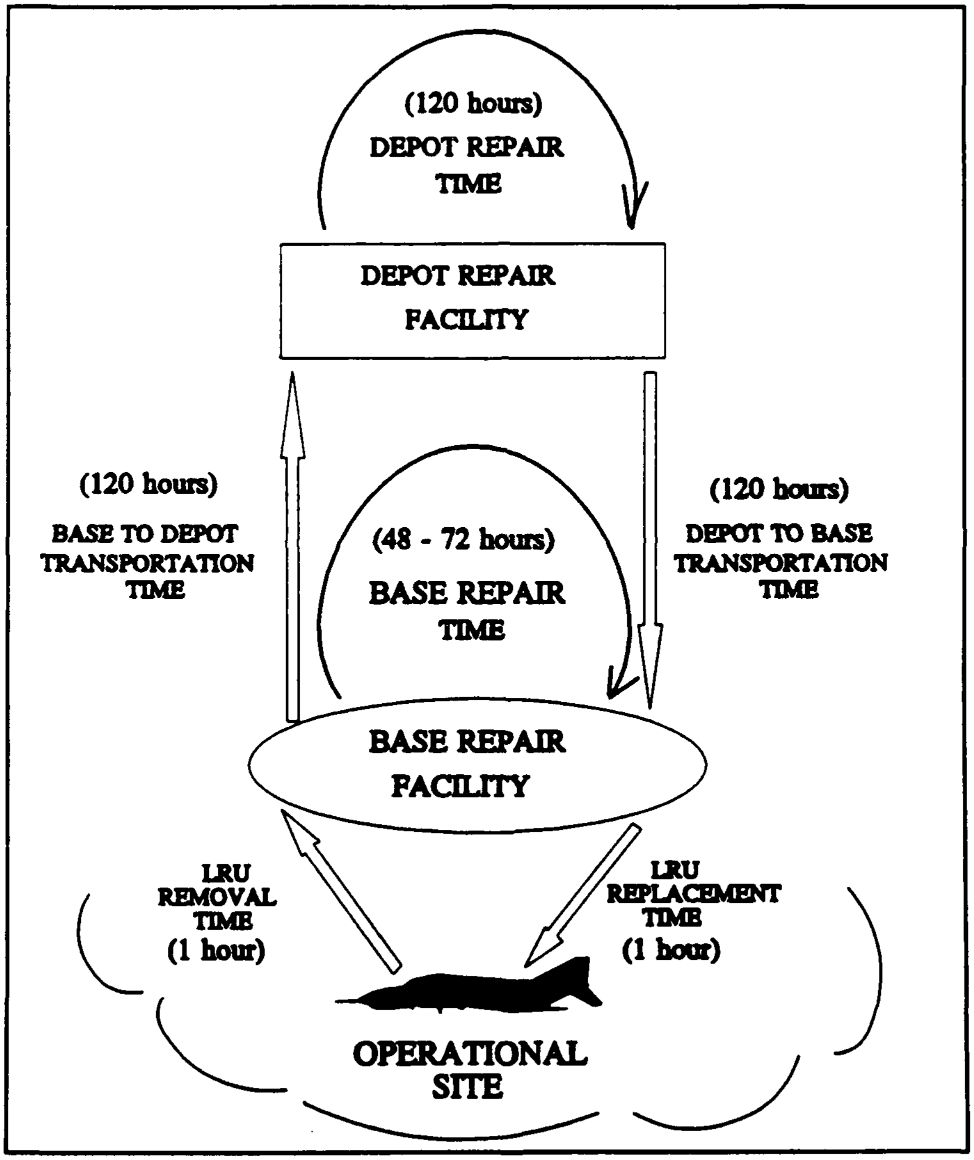

FIGURE 6-1 : MAINTENANCE STRUCTURE FOR THE EXAMPLE.

The logistics support for the aircraft consists of one repair base and a repair depot having the repair turn-around times given in Figure 6-1. The transportation time from base to depot has a value of 120 hours denoted by *TBDTIM* and transportation time from depot to base also has a value of 120 hours denoted by *TDBTIM.*

B. RESULTS FROM ANALYTICAL CALCULATIONS

The wartime period was divided into twenty equal time intervals for the analytical calculations. PC-MATLAB [Ref. 5] was chosen because of its powerful features in matrix manipulation and ease in plotting the results. It was also preferred because it takes much less time to program the procedures as compared to FORTRAN. The exact analytical expressions derived in Chapter III can be expressed in MATLAB almost exactly as they are written in conventional mathematical forms. Two cases were used; the first (Analytical Case One) had unlimited base repair and depot repair capabilities and the second (Analytical Case Two) had no repair capability during the 720-hour wartime period.

Appendix B presents the listing of the program which implement the analytical model for Analytical Case One in PC-MATLAB using the computational steps described in the following subsections. To illustrate the process, the computational details were carried out for one *LRU, LRU-D.* The results of all other *LRUs* were computed in the same manner using PC-MATLAB.

1. Computing the Demand Rates and Repair Flows

The *ns(t)* used in the analytical calculation of the demand rates based on Equation 3.1 is assumed constant at a value of 24. This is a worst-case calculation for the demand rates. The distribution of the failure arrivals for each *LRU* follows a nonhomogeneous Poisson process with only two demand rates based on the utilization rates.

Based on Equation 3.1, the peak demand rate, D_1, for *LRU-D* is

$$D_1 = f \times q \times u(t) \times 24$$

$$= (1/86.9) \times 1 \times 0.2 \times 24 = 0.055 \text{ failures/hour.} \tag{6.1}$$

The lower demand rate, D_2 is

$$D_2 = (1/86.9) \times 1 \times 0.1 \times 24 = 0.028 \text{ failures/hour.} \tag{6.2}$$

From Equation 3.24, D_1 has two components as follows

$$D^b{}_1 = D_1 \,.\, (1 - NRTS) \quad \text{(flow to base)}$$
$$= 0.055 \text{ X } 0.41 = 0.023 \text{ failures/hour,} \tag{6.3}$$

$$D^d{}_1 = D_1 \,.\, NRTS \quad \text{(flow to depot)}$$
$$= 0.055 \text{ X } 0.59 = 0.032 \text{ failures/hour.} \tag{6.4}$$

D_2 also has two components as follows

$$D^b{}_2 = D_2 \,.\, (1 - NRTS) \quad \text{(flow to base)}$$
$$= 0.028 \text{ X } 0.41 = 0.011 \text{ failures/hour,} \tag{6.5}$$

$$D^d{}_2 = D_2 \,.\, NRTS \quad \text{(flow to depot)}$$
$$= 0.028 \text{ X } 0.59 = 0.016 \text{ failures/hour.} \tag{6.6}$$

2. Calculating the Base Repair Pipeline

From Table 6-1, the base repair time for *LRU-D* has a constant value of *BRTIM* = 72 hours. A time chart depicted in Figure 6-2 is used to indicate that the computation of the expected *Base Repair Pipeline,* $\Lambda^b(t)$, for *LRU-D* is time dependent. Equation 3.5 is used for all the computations of $\Lambda^b(t)$ where the term $[1 - G(s,t)]$ has a value of one for all the calculations. This is due to the fact that *BRTIME* is a constant.

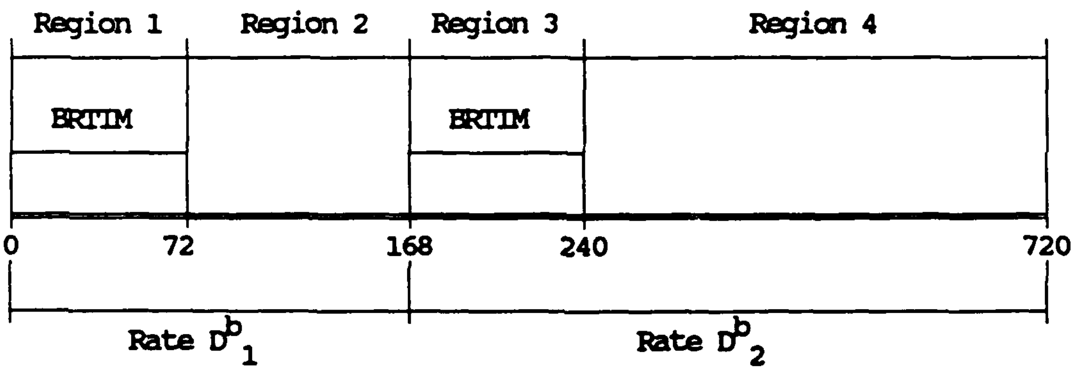

FIGURE 6-2 : TIME CHART FOR COMPUTATION OF THE BASE PIPELINE FOR LRU-D.

In time region 1 where $\{0 < t \leq 72\}$, $\Lambda^b(t)$ increases linearly with time since all the failed units of *LRU-D* are still undergoing repair at the base at a rate of $D^b{}_1$.

Hence,

$$\Lambda^b(t) = \int_0^t D_1^b \, ds = D_1^b \, . \, t = 0.023t. \tag{6.7}$$

In time region 2 where *{72 < t ≤ 168}*, only failed units of *LRU-D* that are in repair from time *{t-72}* to time t are still in repair at time t at a rate of $D^b{}_1$. So,

$$\Lambda^b(t) = \int_{t-72}^{t} D_1^b \, ds = D_1^b \, . \, 72 = 1.631 \, . \tag{6.8}$$

For time region 3 where *{168 < t ≤ (168+72)}*, the demand rate changes to D_2^b. However, there is still a residual number of units in repair resulting from the influence of D_1^b. Therefore,

$$\begin{aligned} \Lambda^b(t) &= \int_{t}^{168+72} D_1^b \, ds + \int_{168}^{t} D_2^b \, ds \\ &= D_1^b.(168+72 - t) + D_2^b.(t - 168) = 3.672 - 0.012t \, . \end{aligned} \tag{6.9}$$

In time region 4 where *{168+72 < t ≤ 720}*, only D_2^b is causing failures of *LRU-D.* In the same manner as Equation 6.8, only failed units that are in repair from time *{t-72}* to time t are still in repair at time t. Therefore,

$$\Lambda^d(t) = \int_{t-72}^{t} D_2^b \, ds = D_2^b \, . \, 72 = 0.792 \, . \tag{6.10}$$

3. Calculating the Total Depot Pipeline

The combined turn-around time at the depot is also a constant with a value equal to the sum of

TBDTIM + *TDBTIM* + *DRTIM* = 120 + 120 + 120 = 360 hours.

Using the same approach described in the previous subsection, a time chart is also used to illustrate the time dependent computation of the *Total Depot Pipeline* or the equivalent term $\Lambda^u(t)$ which was specified by Equation 28. This is shown in Figure 6-3.

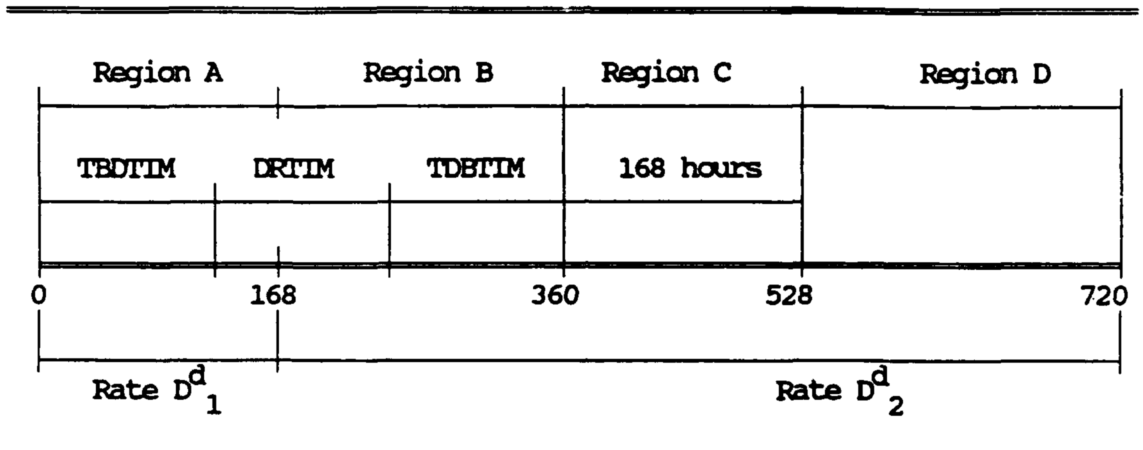

FIGURE 6-3 : TIME CHART FOR COMPUTATION OF THE TOTAL DEPOT PIPELINE FOR LRU-D.

In time region A where $\{0 < t \leq 168\}$, $\Lambda^u(t)$ increases linearly with time since all the failed units of *LRU-D* is still being transported to or in repair at the depot under the peak demand rate D^d_1. Hence,

$$\Lambda^u(t) \;=\; \int_0^t D_1^d \, ds \;=\; D_1^d \cdot t \;. \tag{6.11}$$

In time region B where $\{168 < t \leq 360\}$, a portion of $\Lambda^u(t)$ is increasing linearly with time under the influence of D^d_2. The other portion due to D^d_1 has to wait for repair with a time constant of 168 hours since no depot repair is possible.

Therefore,

$$\Lambda^u(t) = \int_0^{168} D_1^d \, ds + \int_{168}^{t} D_2^d \, ds$$

$$= D_1^d.168 + D_2^d.(t - 168) = 2.688 + 0.016t \,. \tag{6.12}$$

For time region C where *{360 < t ≤ 360+168}*, the portion of $\Lambda^u(t)$ under the influence of $D^d{}_2$ is still waiting for repair since it only started to accumulate failures from 168 hours onwards. On the other hand, depot repair is now possible for other portion due to $D^d{}_1$ which started from time zero but, like before, there is a residue number under repair for a time period of *{360+168 - t}*. So,

$$\Lambda^u(t) = \int_t^{360+168} D_1^d \, ds + \int_{168}^{t} D_2^d \, ds$$

$$= Dw1.(360+168 - t) + D_2^d.(t - 168) = 14.208 - 0.016t \,. \tag{6.13}$$

For time region D where *{360+168 < t ≤ 720}*, only $D_2{}^d$ is causing failures of the LRU-D. Again only failed units of *LRU-D* from time *{t-360}* to time *t* will still in repair at time *t*. Therefore,

$$\Lambda^d(t) = \int_{t-360}^{t} D_2^d \, ds = D_2^d \,.\, 360 = 5.76 \,. \tag{6.14}$$

4. Calculating the Total Pipeline

Since there are only one base repair and one depot repair with no inventory stocking at the depot, the *Total Pipeline* is just the sum of the *Base Repair Pipeline* and the *Total Depot Pipeline* which is

$$\Lambda(t) = \Lambda^{b}(t) + \Lambda^{u}(t) \; . \tag{6.15}$$

The results are discussed later in section E.

5. Calculating Expected Number of Backorders

After $\Lambda(t)$ were computed for each *LRU* (i.e., $\Lambda_i(t)$),the expected number of backorders, $EB_i(t)$, were calculated based on Equation 3.35. The results are discussed later in section E.

6. Calculating Operational Availability

The Operational Availability, $Ao(t)$, calculation without cannibalization is based on Equation 3.39, and with cannibalization based on Equation 3.47, denoted by $Ao_c(t)$. Analyses of these results are presented later in section E.

7. Calculating ENMCS(t)

The $ENMCS(t)$ calculation without cannibalization is based on Equation 3.48 and $ENMCS_c(t)$ for cannibalization is based on Equation 3.46. Analyses of these results are presented later in section E.

8. Calculating P(ENMCS ≤ NMCS)

Although not to be used for comparison with the simulation model, the alternate *MMOE, P(ENMCS ≤ NMCS),* has the same calculation for both cannibalization policies, based on Equation 3.44. The target *NMCS* is set at a value of 4. However there will be no analysis of these results.

9. Computations for Analytical Case Two

The computational steps for Analytical Case Two are straight forward since each LRU has only one pipeline which increases linearly with time. Essentially, the pipeline is calculated by multiplying a demand rate and the time period of time zero to time t. The peak demand rate D_1, computed using Equation 6.1, is used for any time t within time region $\{0 < t \leq 168\}$ and for time region $\{168 < t \leq 720\}$, the lower demand rate D_2, based on Equation 6.2, is used. Graphical and numerical results of the *Total Pipeline* and the *Backorders* for this case are almost the same as those obtained under the OPUS-8 Approximation (see Figures 6-6 and 6-13). Therefore these results are not repeated here.

C. RESULTS FROM SIMULATION RUNS

Five simulation variations were set up for the example. These variations use an input format similar to the one shown in Table 6-2 which list the specific input parameters for Simulation Variation Four.

TABLE 6-2 : INPUT DATA FORMAT FOR WSM SIMULATION RUNS				
3	= random seed for base mean repair time			
5	= random seed for interarrival time of failures			
2	= random seed for ratio of NRTS			
1	= random seed for MTTR (fault isolation time)			
4	= random seed for depot mean repair time			
5000	= number of replications for each simulation			
20	= number of time intervals (36 hours per interval)			
5	= number of repair stations at the base			
24	= number of aircraft initially deployed for war			
120	= transportation time from base to depot			
120	= transportation time from depot to base			
2	= aircraft utilization rates :	FROM	TO (hours)	RATE
		0	168	0.2
		168	720	0.1
4	= maximum number of aircraft allowed to be cannibalized			
0.1	= Tolerance for statistical convergence			

Table 6-3 provides a summary of the five variations and their differences. Details of each variation are given in the following subsections. The results from each variation are discussed later in section E.

TABLE 6-3: SUMMARY OF DIFFERENCES FOR THE FIVE SIMULATION VARIATIONS

SIMULATION RUNS	MAIN DESCRIPTIONS
Variation One	- Exponential interarrival times for failures of all *LRU*. - Deterministic values for all repair and transportation times. - No repair capability. - No cannibalization allowed.
Variation Two	- Same as Variation One except that there is unlimited repair capability at the base and the depot.
Variation Three	- Same as Variation Two except that repair times are now made exponential. - cannibalization policy is used.
Variation Four	- Same as Variation Three except that the base repair resources are limited. - *FCFS* repair priority is used.
Variation Five	- Same as Variation Four except that *FCFS* is replaced by *LAIF* repair priority.

1. Simulation Variation One

This variation is the simplest simulation case of all the five since no repair is allowed. The purpose of this variation was to verify the adequacy of using OPUS-8 as an approximation to the WSM. This variation also helped to verify the algorithms of the simulation model. All the input parameters except for the exponential random generation of failure arrivals of the *LRUs* were modeled as deterministic to match the characteristics of OPUS-8 input requirements. This also meant that $ns(t)$ is kept constant for the computation of $D(t)$. In the simulation, a failed *LRU* caused the system to be *NMC*. Then, if there were spares available, the *NMC* system was made

MC. When no more spares were left, the systems became *NMC* and when all the twenty-four aircraft were made *NMC*, the replication was considered complete and the simulation went to the next replication.

2. Simulation Variation Two

The purpose of Variation Two was to ascertain whether the results by simulation were close to the results of Analytical Case One when the input parameters were kept the same for both. To conform with the assumptions of Analytical Case One, the repair resources at both the base and the depot were assumed to be unlimited and the policy of no cannibalization was adopted. No distribution assumptions were possible for the failure arrivals of the *LRUs* since the value of $D(t)$ was expected to fluctuate widely because $ns(t)$ was allowed to vary during the simulation. Indeed, these factors caused differences in results between Analytical Case One and Variation Two. Except for the above, the characteristics of the other inputs are the same as in Variation One.

3. Simulation Variation Three

Variation Three has the same considerations as Variation Two except that repair times at the repair base and repair depot were generated from exponential distributions. Also, the policy of cannibalization is implemented.

4. Simulation Variation Four

Variation Four extends Variation Three to incorporate the realistic problem of limited repair resources during an intense conflict period. The simulation

constrained the repair base to seven repair stations but the repair depot is unlimited. *FCFS* repair priority was used in this variation.

5. Simulation Variation Five

The last variation examines the effect of a different repair priority policy than Variation Four. Here the *LAIF* repair priority is implemented; defective *LRUs* with the largest number of backorders have first priority in the base repair queue.

D. RESULTS FROM OPUS-8 APPROXIMATION

It was found that the OPUS-8's sustainability option provided an approximate result for the WSM only under the condition that there is no repair capability during the anticipated wartime period. However, as explained in Chapter V, a number of runs were required to model the anticipated wartime period since each OPUS-8 run can only provide the results for one time period and one utilization rate. In the example, where 20 time intervals were used to divide the wartime period, the first run presented the time period from time zero to the ending time of the first time interval, the second also started from zero but ended at the ending time of the second interval, and so on up to the twentieth run. In this way, the *Total Pipeline* increased with the run number since no repair was allowed. The approximation also required that the utilization rate for each time period be weighted by the amount of time spent in the first and second utilization rates.

The required number of runs and the corresponding time period and utilization rate are shown in Table 6-4. Except for these two parameters, all other

input parameters were the same for the twenty OPUS-8 runs. The input format and data requirements for the OPUS-8's sustainability option are provided in Appendix E only for run number 10 to illustrate the approximation process. Results are presented later in section E.

TABLE 6-4 : OPUS-8 RUNS TO APPROXIMATE WSM.

RUN #	LENGTH OF PERIOD	WEIGHTED UTILIZATION
1.	36	0.2
2.	72	0.2
3.	108	0.2
4.	144	0.2
5.	180	0.1933
6.	216	0.1777
7.	252	0.1667
8.	288	0.1583
9.	324	0.1519
10.	360	0.1467
11.	396	0.1424
12.	432	0.1389
13.	468	0.1359
14.	504	0.1333
15.	540	0.1311
16.	576	0.1292
17.	612	0.1275
18.	648	0.1259
19.	684	0.1246
20.	720	0.1233

E. ANALYSES OF RESULTS

This section shows the pertinent results obtained from the three methods described in the previous sections of this chapter and specifically gives the analyses of the outcomes on the *Total Pipeline, Backorders, Ao(t),* and *ENMCS(t)*.

To enhance the analyses, the following legends are used in all the graphs presented this section:

- **O :** OPUS-8 Approximation
- **A1 :** Analytical Case One without Cannibalization
- **A1c:** Analytical Case One with Cannibalization
- **A2 :** Analytical Case Two
- **S1 :** Simulation Variation One
- **S2 :** Simulation Variation Two
- **S3 :** Simulation Variation Three
- **S4 :** Simulation Variation Four
- **S5 :** Simulation Variation Five.

1. Analyses on Total Pipeline

For the purpose of selecting a few *LRUs* for in-depth analysis, the *Total Pipeline* results for all the ten *LRUs* for the analytical model, the simulation model and OPUS-8 Approximation are first examined. Figures 6-4, 6-5 and 6-6 provides the these results graphical for Analytical Case One (A1), Simulation Variation Four (S4)

and OPUS-8 Approximation (O) respectively. The corresponding numerical values for A2 and S4 are tabulated in Tables C-1 and D-1. Numerical results from O are identical to A2.

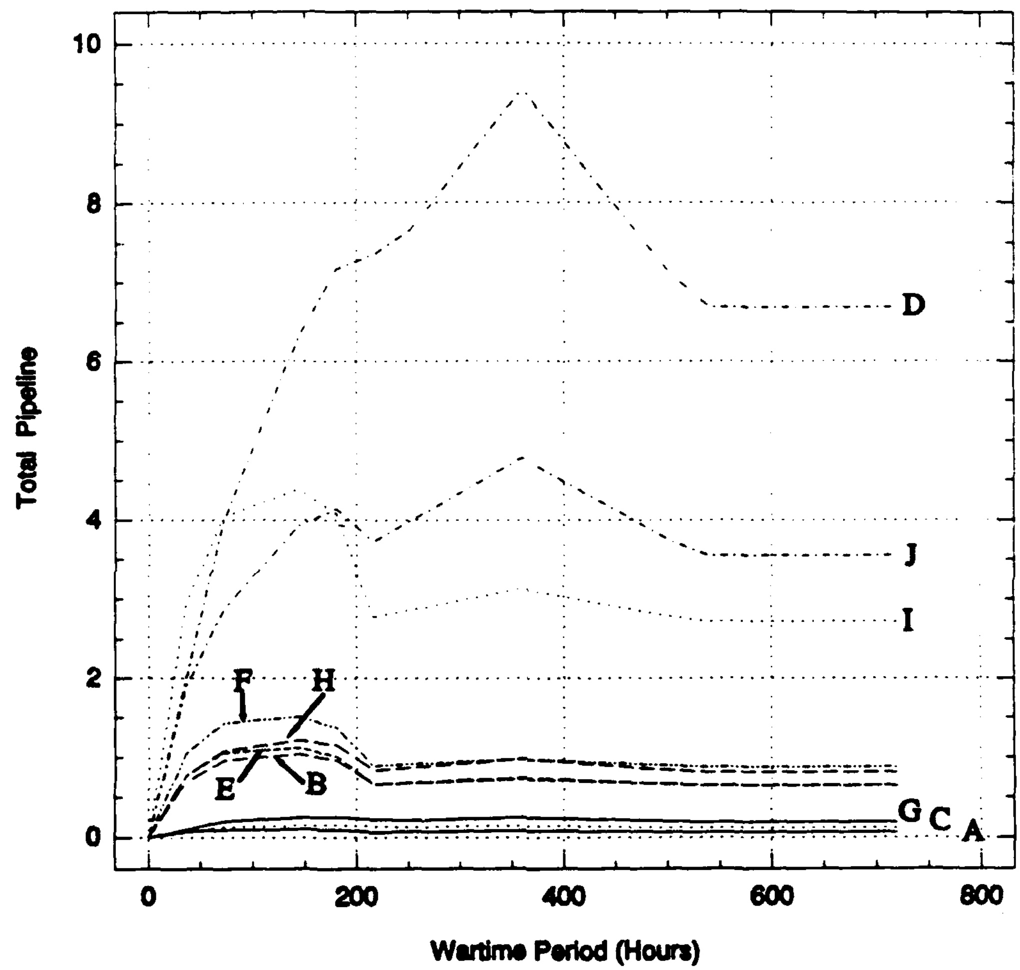

FIGURE 6-4: RESULTS OF TOTAL PIPELINE FOR ALL LRUS UNDER ANALYTICAL CASE ONE.

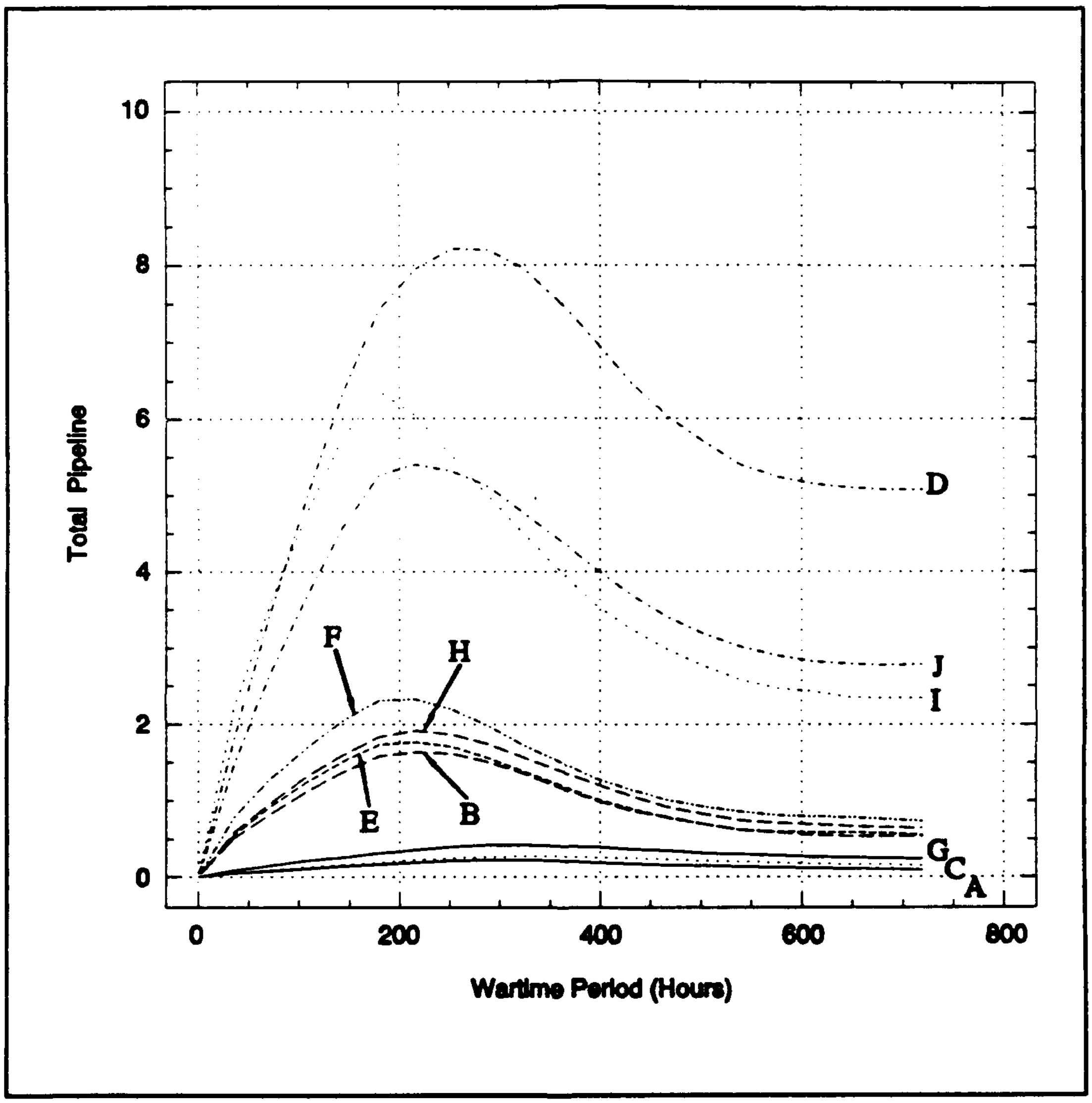

FIGURE 6-5: RESULTS OF TOTAL PIPELINE FOR ALL LRUS FROM SIMULATION VARIATION FOUR.

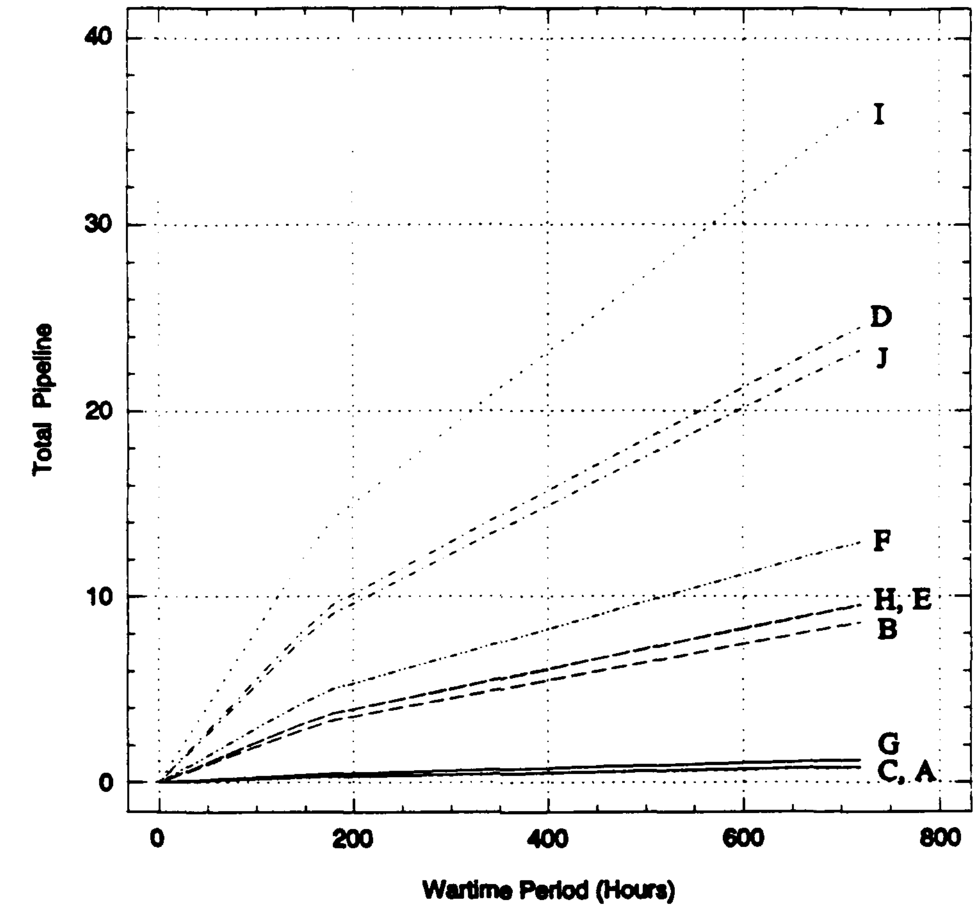

FIGURE 6-6: RESULTS OF TOTAL PIPELINE FOR ALL LRUS FROM THE OPUS-8 APPROXIMATION.

LRU-D, LRU-I and *LRU-J* are observed to have the largest values. As more evident in Figure 6-4, the *Total Pipelines* of *LRU-D* and *LRU-I* peaked at quite different times and displayed other contrasting characteristics. Also, these two LRUs

displayed other different behavior in their Backorders results, which are shown in Figures 6-11 and 6-12 of the next subsection. Therefore from now on, only these two *LRUs* are further analyzed.

First we consider the input characteristics of these *LRUs* for some explanations of their differing results. As given in Table 6-1 under the *NRTS* column, *LRU-D* had a high proportion *(59%)* of its failed units flowing to the depot for repair whereas the proportion for *LRU-I* is only *6%*. Other differences are; *MTBF* of 86.9 hours for *LRU-D* and 58.8 hours for *LRU-D*, mean base repair time with *LRU-D* needing 72 hours and *LRU-I* needing 48 hours, and initial inventory levels of 5 and 6, respectively. The rest of the inputs were the same.

Figure 6-7 shows the results of the *Total Pipeline* for *LRU-D* from the three versions of the WSM under the condition that there was no repair capability during the whole anticipated wartime period. This can be considered the worst case scenario in which all the repair resources were severed by the enemy at the onset of war. All the results agree extremely well. It is understandable that all the results of the *Total Pipeline* are monotonically increasing since when repair was not allowed, more and more failed units of *LRU-D* became unserviceable as the war progressed. The extremely close outcomes suggest that both the analytical model and the simulation model are sound and also that OPUS-8 Approximation is a reasonable approach.

Figure 6-8 shows similar trends for *LRU-I* although the ordinate values are larger. This was mainly due to *LRU-I* having a lower *MTBF* and, as a consequence,

a higher inherent failure rate than *LRU-D*. The other differing inputs mentioned earlier had no effects since no repair was involved.

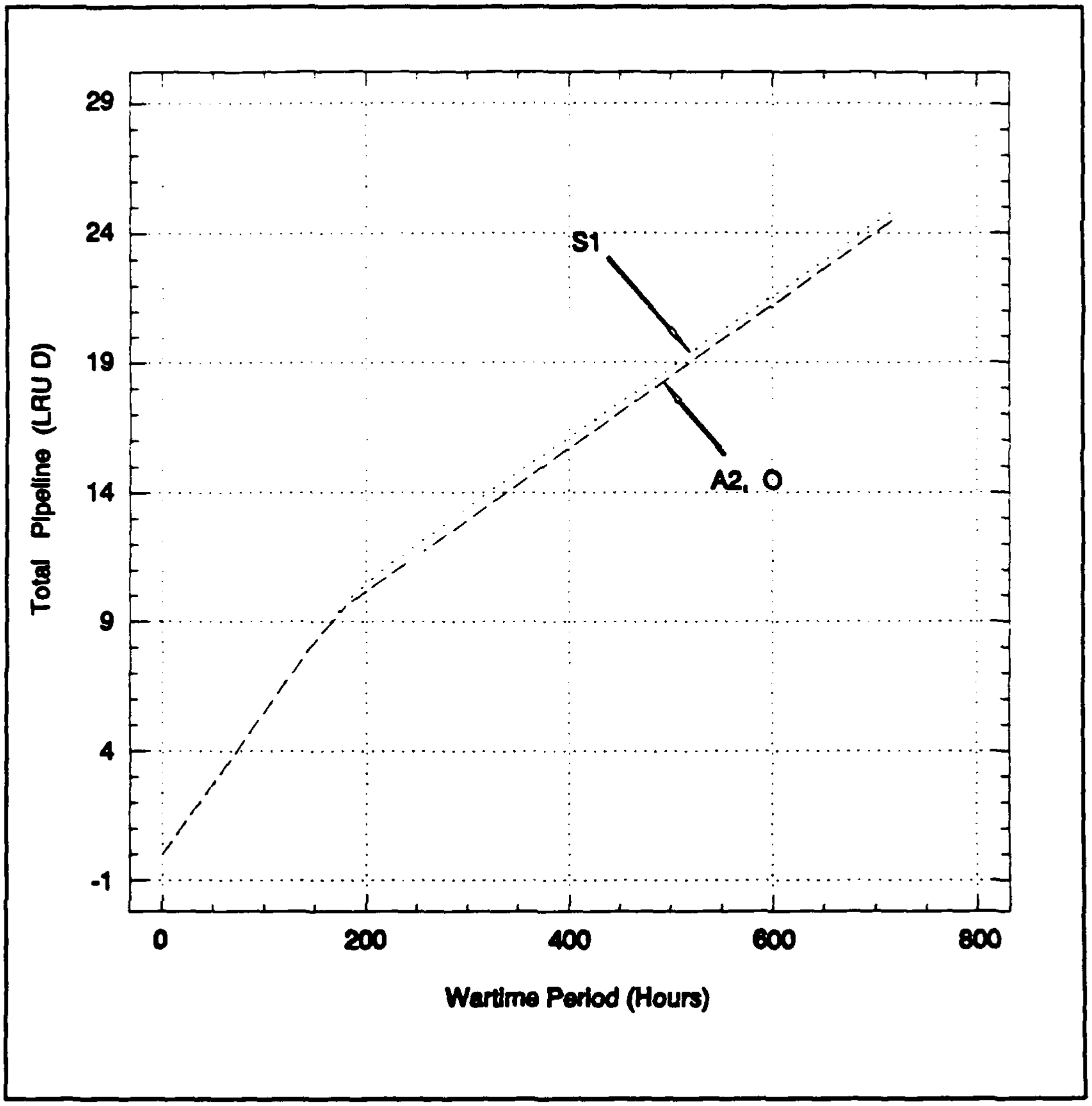

FIGURE 6-7: RESULTS OF TOTAL PIPELINE FOR LRU-D FOR MODEL VARIATIONS HAVING NO REPAIR CAPABILITY.

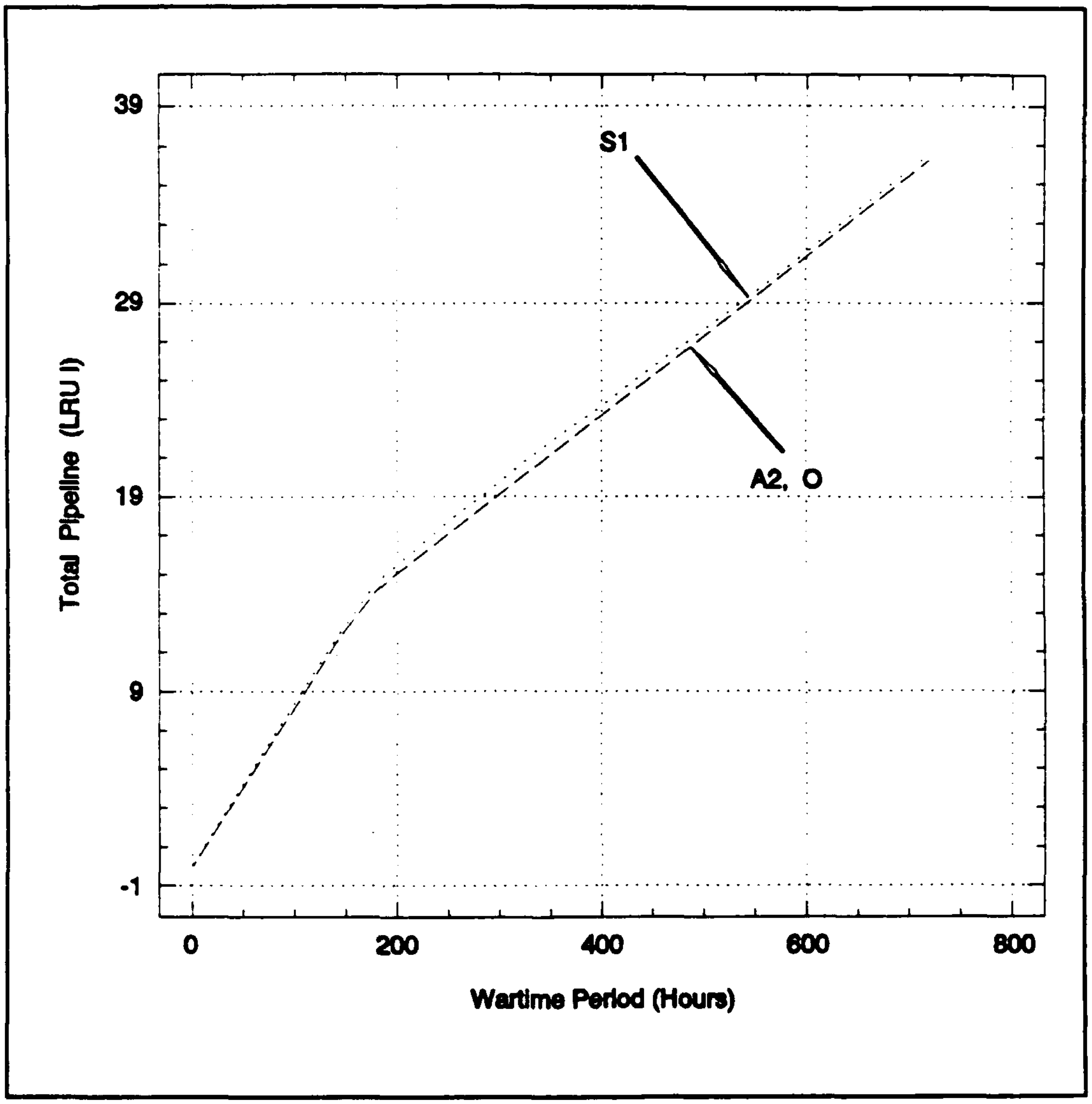

FIGURE 6-8: RESULTS OF TOTAL PIPELINE FOR LRU-I FOR MODEL VARIATIONS WITH NO REPAIR CAPABILITY.

More interesting trends were observed for those model variations having a repair capability. Figure 6-9 shows five different results for the *Total Pipeline* of *LRU-D*. A1 and S2 had more abrupt changes than the others. This is due to the

deterministic repair times being used at both the base and the depot. When exponential repair times were used the results showed more gradual changes as evident from S3, S4 **and** S5.

The highest value of *Total Pipeline* came from **A1** which peaked at a value of 9.42 at 360 hours. The high proportion of failures flowing to the depot for repair is probably the dominant cause of this peak since the total depot turn-around time was 360 hours. **S2** had a similar profile for the same reason. However, for **S4** and **S5** the limited repair resources at the repair base result in failures having to queue for these resources. Their peaks occurred consistently at 252 hours which is an indication of a repair "bottleneck" caused by the base repair queue. The number in the queue is observed to dominate the number in the repair depot.

Among the three cases with unlimited base repair capability, the lowest range of values between $t=0$ and $t=500$ hours was observed when the policy of cannibalization was adopted for **S3**. This indicates that the cannibalization policy is an effective means of minimizing the number of demands for spares since good parts from *NMC* systems can be cannibalized as spares. For the two cases with limited base repair capability and under the policy of no cannibalization for *LRU-D*, the *LAIF* priority (S5) has a better result than the *FCFS* priority (S4) although the difference was not as significant as the cannibalization vs. no cannibalization comparison. This is a consistent result since the squadron is expected to sustain better when its worst *LRUs* are repaired first.

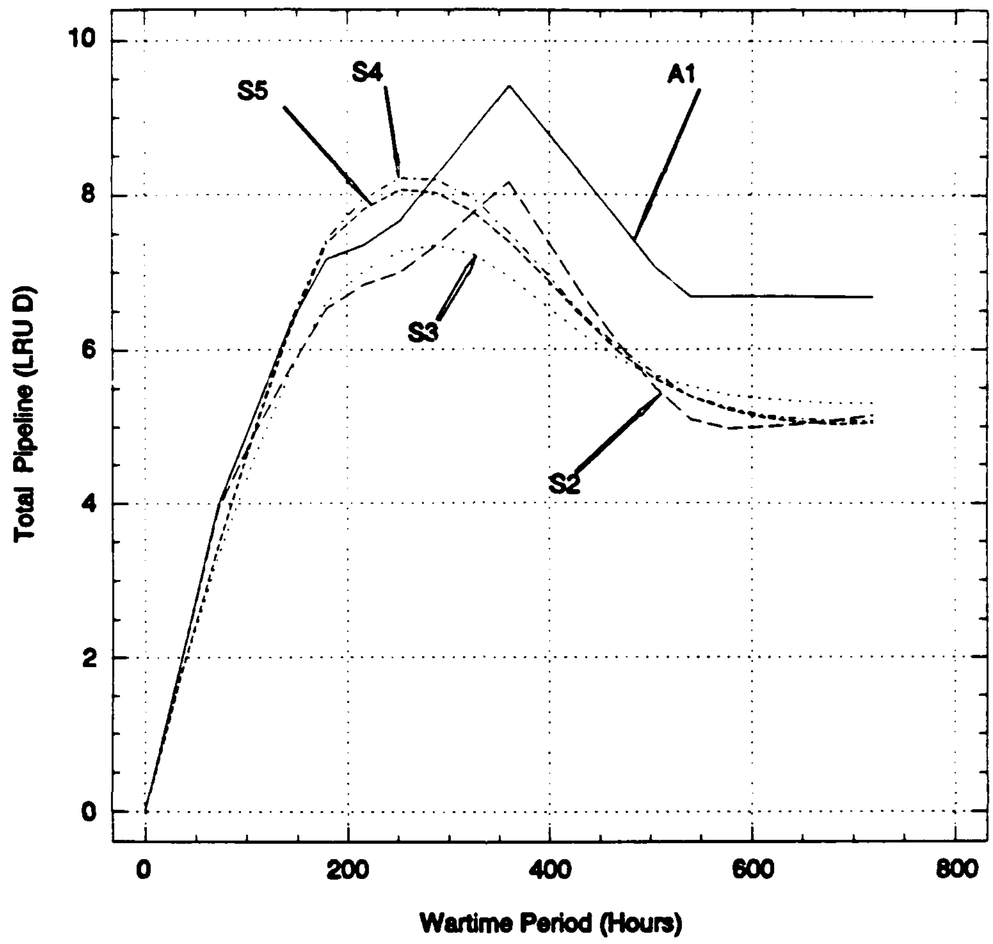

FIGURE 6-9: RESULTS OF TOTAL PIPELINE FOR LRU-D FOR MODEL VARIATIONS WITH REPAIR CAPABILITY.

On the whole for the simulation cases, S3 has the lowest Total Pipeline which is evident from the fact that there was unlimited repair capability and parts can be cannibalized as spares.

All the model variations in Figure 6-9 are observed to reach steady-state at about 550 hours. One might say that this is because the lower utilization rate became the dominant rate by this time period. This would also cause the value of *ns(t)* be to become stable. Therefore, when computed from Equation 3.1, the failure arrival rates of the *LRUs* become more constant. It has been shown in Equation 3.4 that when the failure arrival rate of an *LRU* is constant and follows a Poisson process, then the *Total Pipeline* is in steady-state.

Certain results were observed for *LRU-I* that differ from those in *LRU-D*, as shown in Figure 6-10. The most obvious change is that the peak has shifted to 144 hours for **A1** and **S2**. The peak for the other three cases is now at 180 hours. The most convincing reason for this result is the fact that a very high proportion (94%) of *LRU-I* failures went to the base for repair where the demands for spares were greatest at about 168 hours.

Another major change is the value of the peak itself. The highest peak value now belongs to **S4** closely followed by **S5**. The results from **A1** are much lower. These observations are opposite to what were observed for *LRU-D*. These can be explained by the same fact that 94% of all *LRU-I* failures went to the base for repair as compared to *LRU-D*'s 6%. Also **S4** and **S5** had limited number of base repair resources (5 repair stations). Therefore their *Total Pipeline* are inflated by the number in queue at the base repair. **S4** has a larger number than **S5** since the latter used a better repair prioritization policy (i.e., *LAIF*).

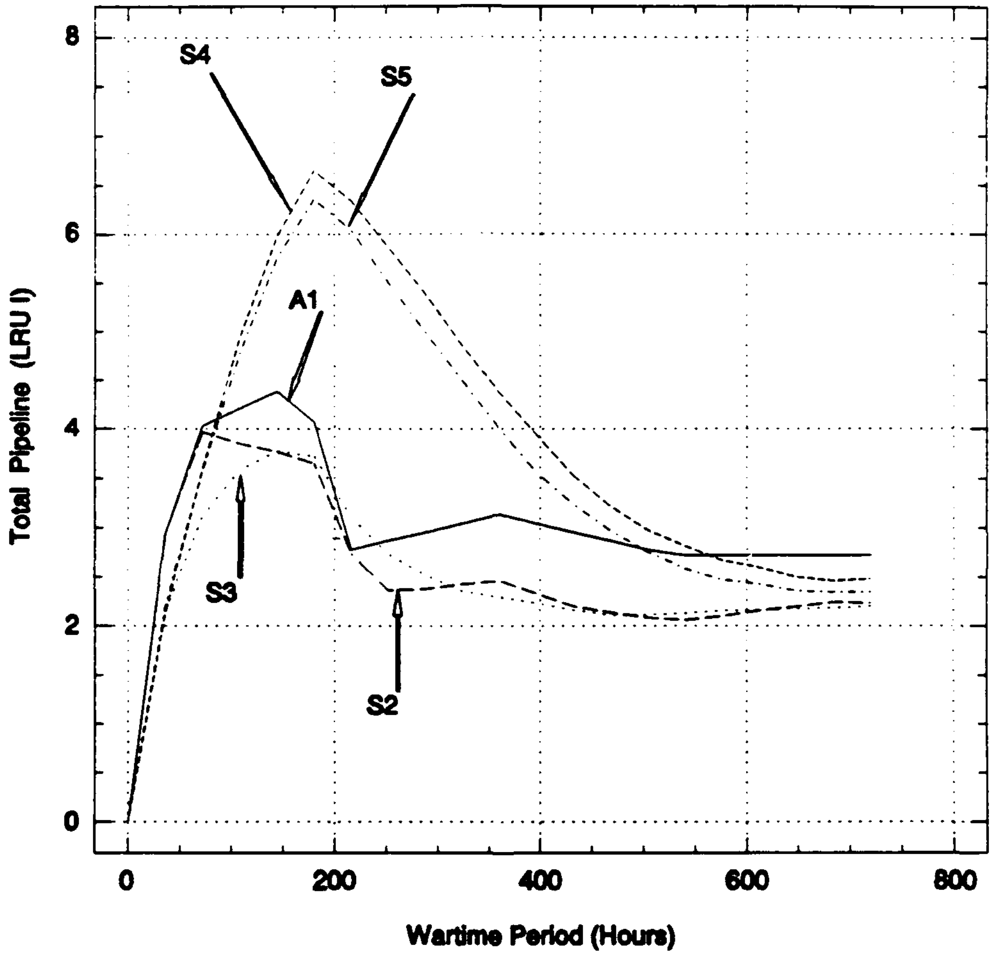

FIGURE 6-10: RESULTS OF TOTAL PIPELINE FOR LRU-I FOR MODEL VARIATIONS WITH REPAIR CAPABILITY.

2. Analyses Of Backorders

As an illustrations of *Backorders* results obtained by the three different versions of WSM , results for all the ten *LRUs* for **A1**, **S4** and **O** are presented graphically in Figures 6-11, 6-12 and 6-13, respectively. The corresponding numerical values for **A1** and **S4** are given in Tables C-2 and D-2. Results from **O** are almost identical to that of **A2** which is shown later in Figure 6-14. In particular for **A2** as shown in Figure 6-11, only *LRU-D* and *LRU-J* have backorders although these two and *LRU-I* experienced high numbers in the *Total Pipeline.* Also in the same graph, no backorders were observed for the other *LRUs* although backorders were observed for all the *LRUs* in the case of **S4** and **O** as shown in the other two figures. The analysis of these differences are discussed below, using only *LRU-D* and *LRU-I* for comparison.

Without repair capability, Figures 6-14 and 6-15 illustrates results of the *Backorders* from model variations **A2**, **S1**, and **O** for *LRU-D* and *LRU-I*, respectively. The closeness of these results also reinforce the suggestions made for the *Total Pipeline* results .

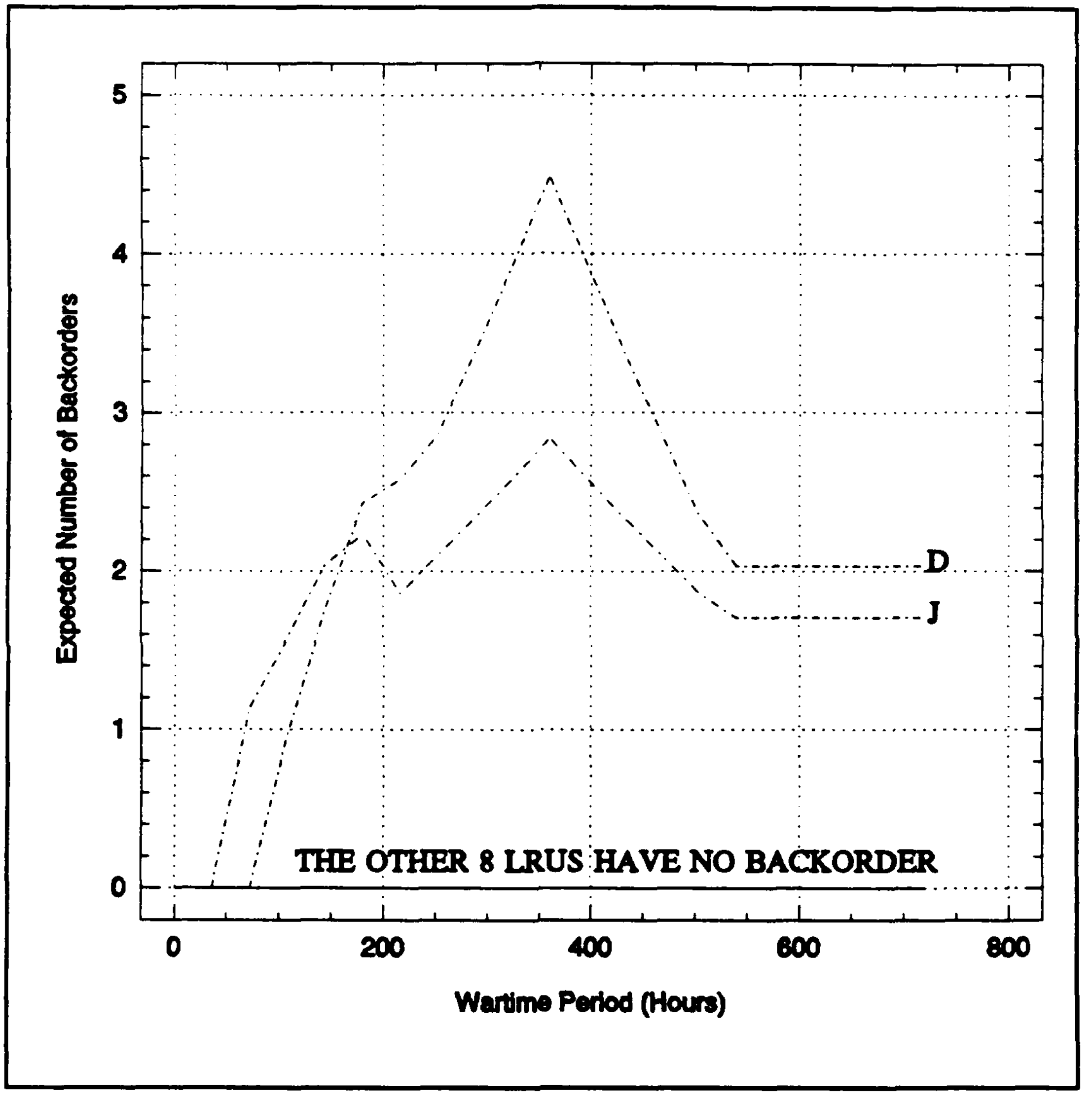

FIGURE 6-11: RESULTS OF BACKORDERS FOR LRUS UNDER ANALYTICAL CASE ONE.

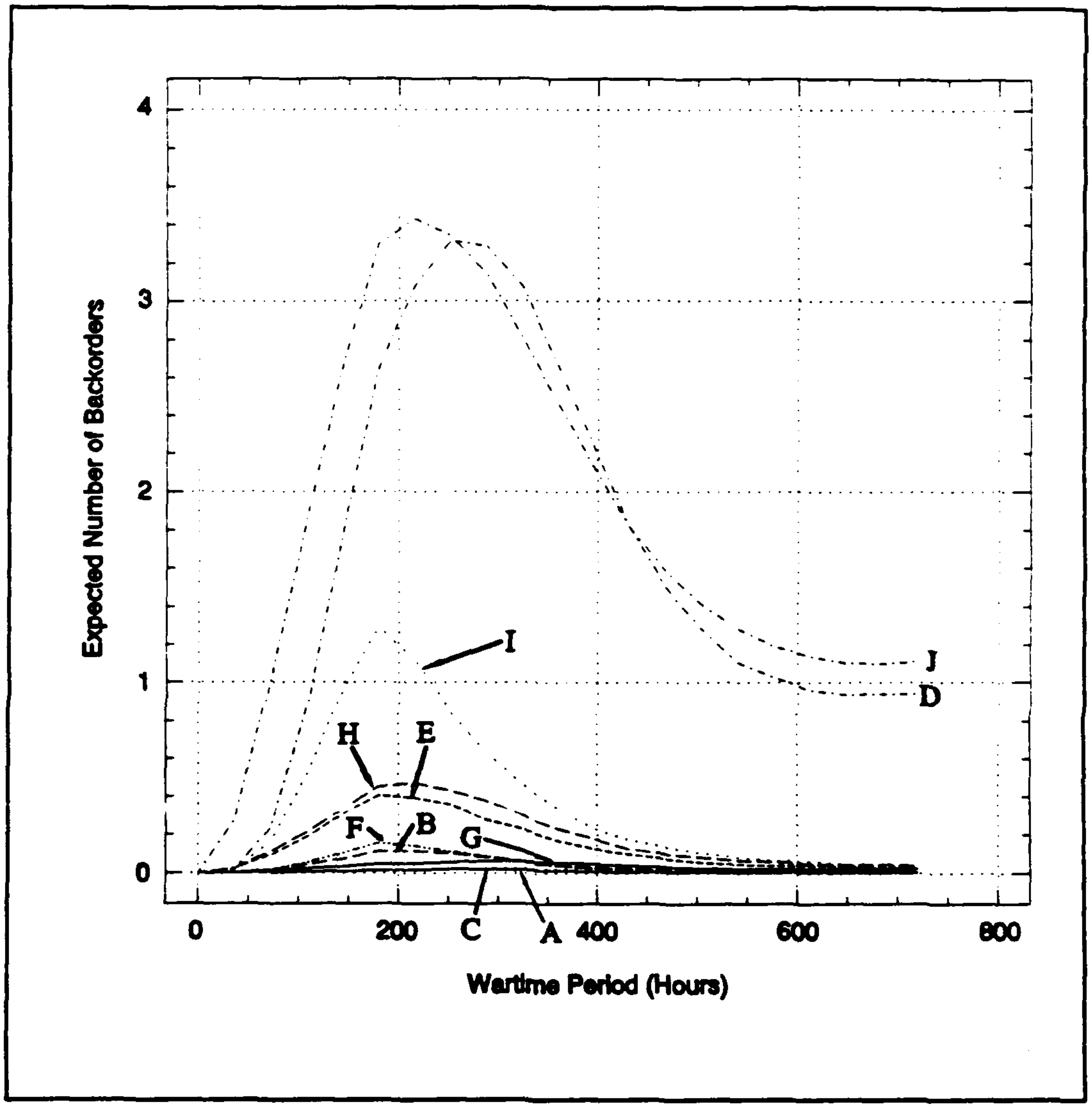

FIGURE 6-12: RESULTS OF BACKORDERS FOR ALL LRUS FROM SIMULATION VARIATION 4.

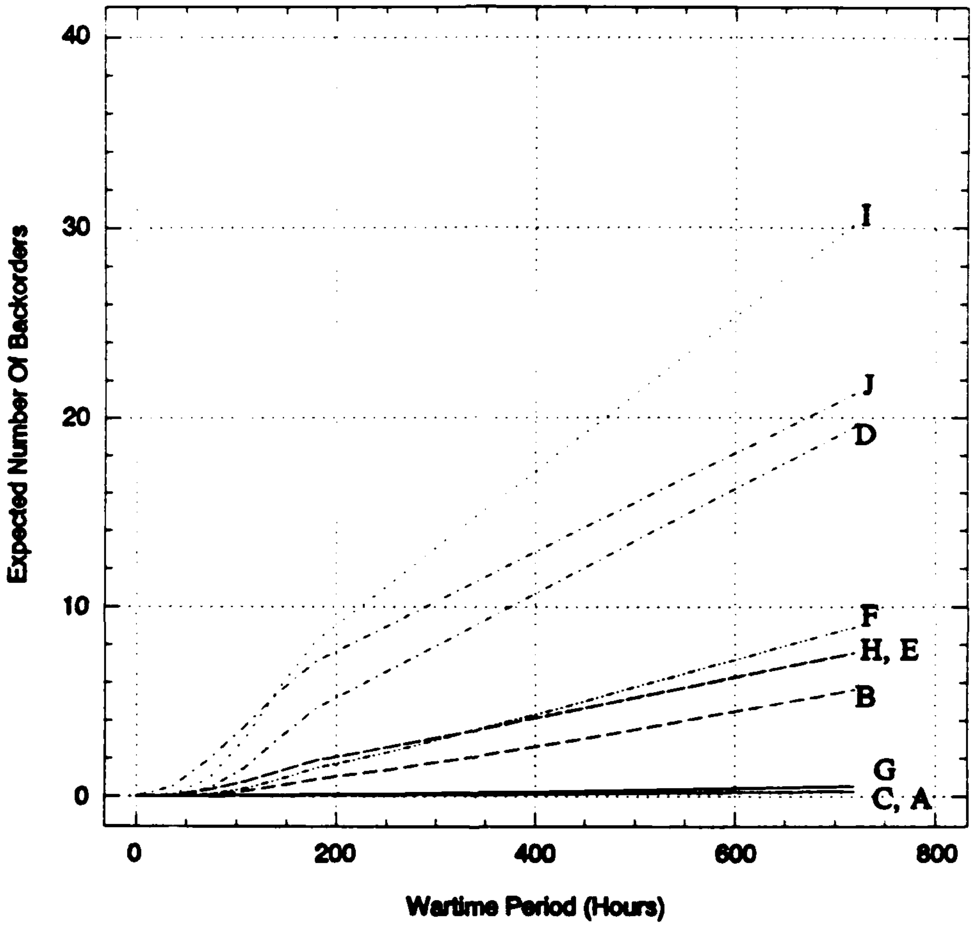

FIGURE 6-13: RESULTS OF BACKORDERS FOR ALL LRUS FROM THE OPUS-8 APPROXIMATION.

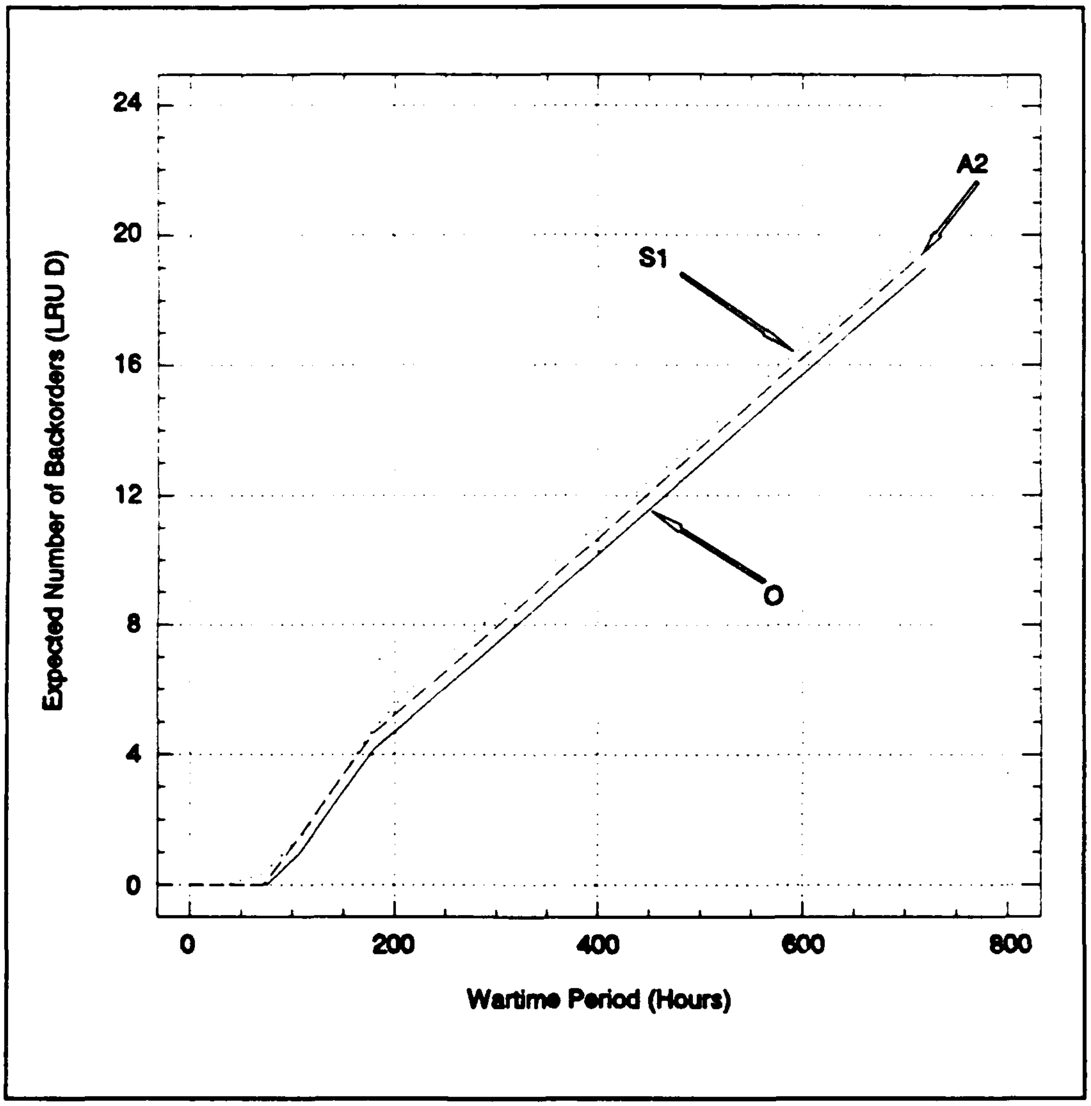

FIGURE 6-14: RESULTS OF BACKORDERS FOR LRU-D FOR MODEL VARIATIONS WITH NO REPAIR CAPABILITY.

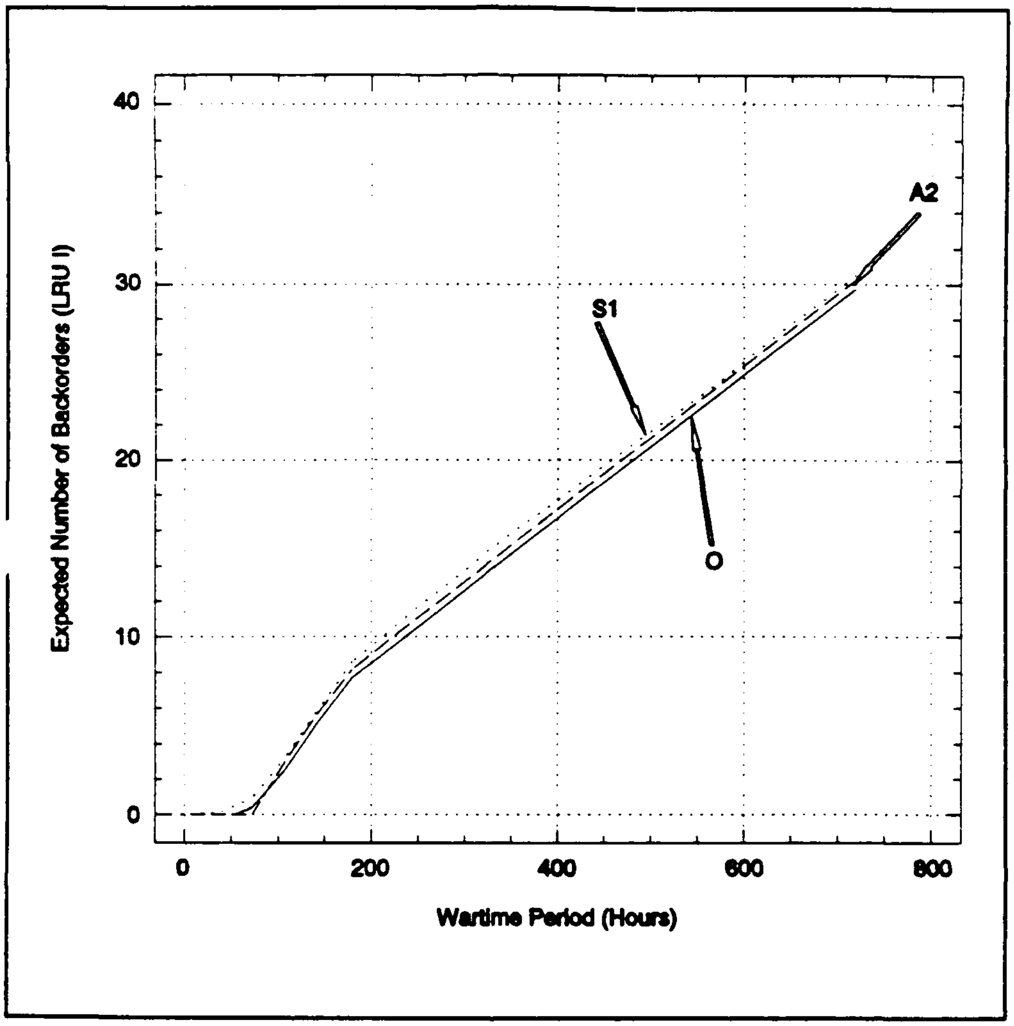

FIGURE 6-15: RESULTS OF BACKORDERS FOR LRU-I FOR MODEL VARIATIONS WITH NO REPAIR CAPABILITY.

As before, more interesting results came from those cases with repair capability. *LRU-D* which has a comparatively large number of units in its *Total Pipeline* as evident from Figure 6-9 of the previous subsection. In fact, A1, S2, S3, S4 and S5 have *Total Pipeline* of at least a value of 5 when time is more than 100

hours. Although *LRU-D* has a comparatively high initial inventory level of 5, these spares are insufficient to meet a *Total Pipeline* of more than 5. As a consequence, *Backorders* are expected to occur early in the wartime period and Figure 6-16 indicates that this is true.

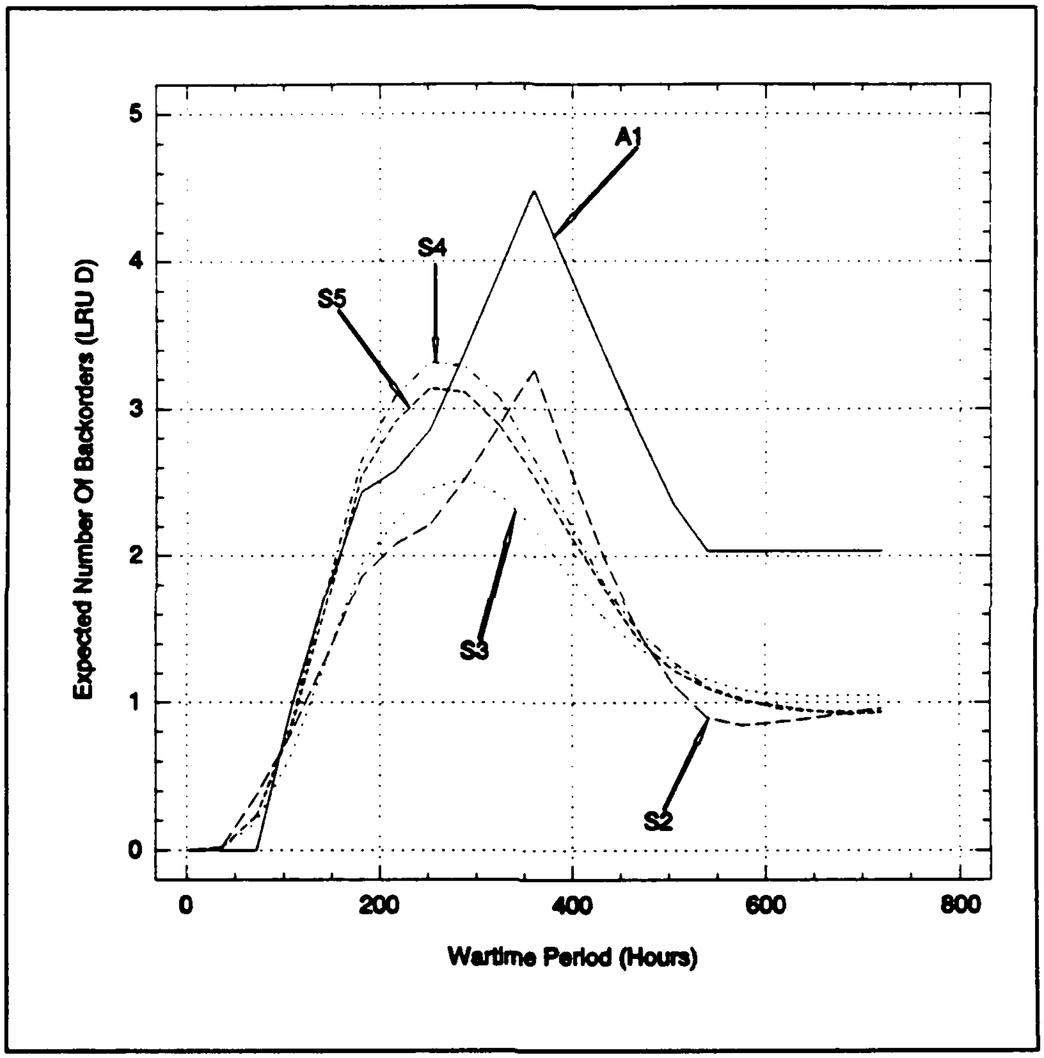

FIGURE 6-16: RESULTS OF BACKORDERS FOR LRU-D FOR MODEL VARIATIONS WITH REPAIR CAPABILITY.

In fact, the *Backorders* for the five cases are higher than for any of other *LRUs*. Other than this, the *Backorders* profiles are similar to the corresponding *Total Pipelines* profiles of *LRU-D* for the same reasons given in the previous subsection.

In the case of *LRU-I*, only S4 and S5 have peak *Total Pipelines* that exceed a value of 6 as indicated in Figure 6-10 of the previous subsection. With an initial inventory level of 6, the *Backorders* results depicted in Figure 6-17 showed the expected outcome that only these two cases have significant *Backorders* while the other cases displayed negligible *Backorders*. Hence, these results suggest that for any *LRU*, the *Backorders* depends mainly on the *Total Pipeline* and the initial inventory level.

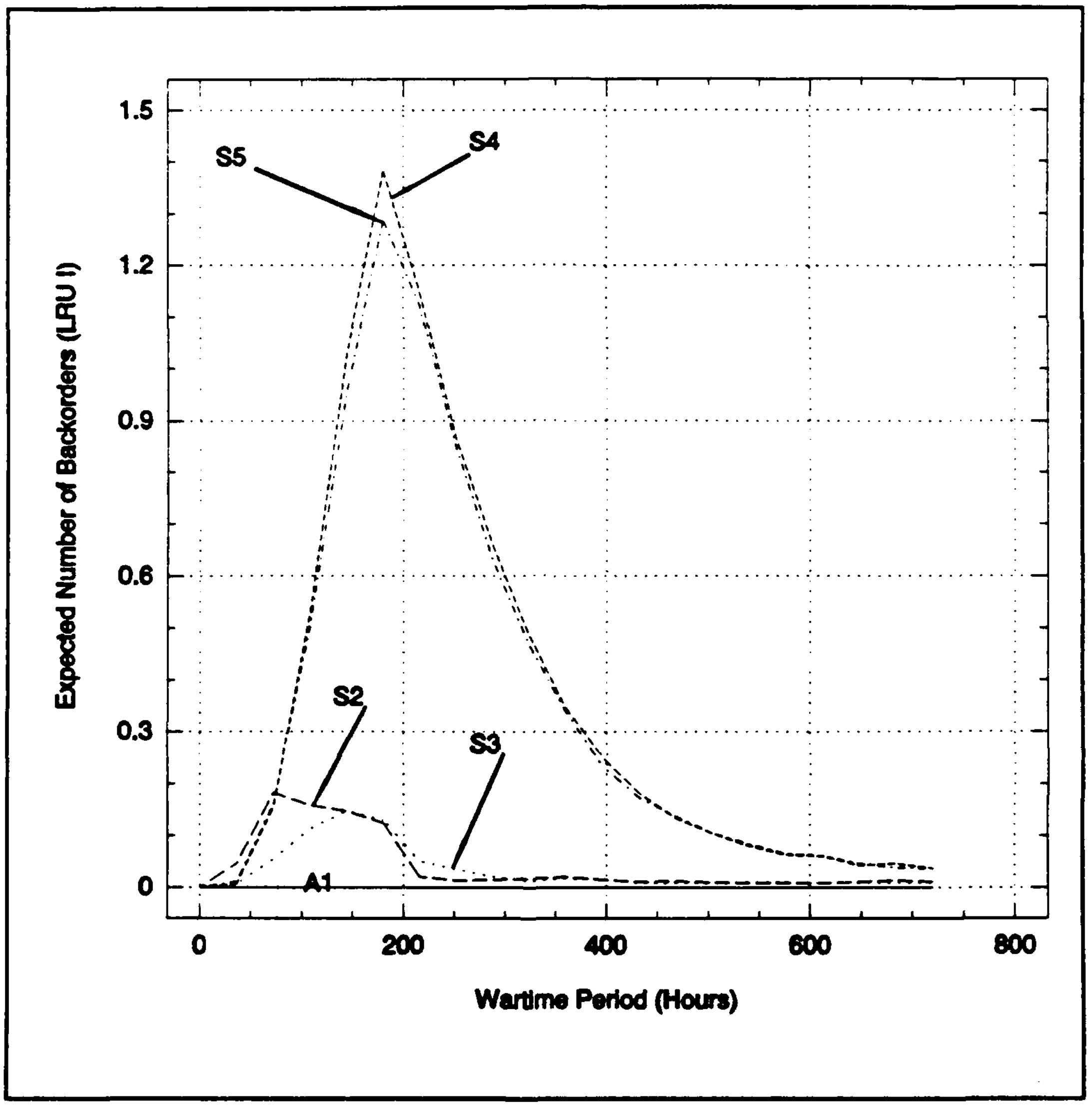

FIGURE 6-17: RESULTS OF BACKORDERS FOR LRU-I FOR MODEL VARIATIONS WITH REPAIR CAPABILITY.

3. Analyses on Operational Availability, Ao(t).

Under the policies of no repair capability and no cannibalization, the *Ao(t)* results obtained from **A2**, **S1** and **O** are shown in Figure 6-18. These *Ao(t)* results are observed to reach a value of 0.0 at about 450 hours from the start of the war. These results reflect the fact that more and more deployed systems became *NMC* as time progressed under the conditions of no repair and limited amount of initial inventory supply. Also, the results from the analytical model and the simulation model are quite similar up to about 150 hours. The values from **S1** change more smoothly since they are average values weighted by at least 5000 replications. On the other hand, the curve produced by the OPUS-8 Approximation has a more uneven profile and an offset as compared to the other two. The unevenness is understandable since the results were obtained from an approximation method of twenty OPUS-8 runs. Nevertheless, they all converged to a value of zero at about 450 hours. This is a good indication that the three versions of the WSM have similar asymptotic characteristics.

The *Ao(t)* results for model variations with repair capability are depicted in Figure 6-19. An additional result from case **A1** is obtained by introducing a cannibalization policy and using Equation 3.47.

The lowest *Ao(t)* value for all six cases occurs at between 300 and 400 hours from the start of the war. This can be explained by the fact that *LRU-D and LRU-J,* which have the most *Backorders* among all the *LRUs* (see Figures 6-11 and

6-12) peaked at between 200 and 350 hours. The possible causes of this can be attributed to effects such as

a. a long total depot turnaround time of 360 hours for all cases,

b. limited base repair resources for **S4** and **S5**, and

c. relatively high demands for spares but insufficient inventory level for *LRU-J*.

Therefore, it can be deduced that short repair turnaround times plus adequate repair resources, reliable *LRUs* and sufficient inventory stocks for these *LRUs* are fundamental requirements of a logistics support policy to sustain a military capability during the wartime period.

The results also indicate that **A1** with cannibalization, or **A1c**, has a better outcome than not allowing cannibalization.

As observed previously in *Total Pipeline* and *Backorders,* steady-state characteristics for all simulation cases are also noticed near the end of the wartime period. Analytical cases stabilized earlier at about 550 hours which is similar to the results for *Backorders.*

Overall, the *Ao(t)* results of all the cases are quite close in value. This is a good indication that the formulae derived for the analytical model and the algorithms used for the simulation model are comparable. However, more complex examples should be investigated to draw more definitive conclusions and this is recommended as a subject for further development for the WSM.

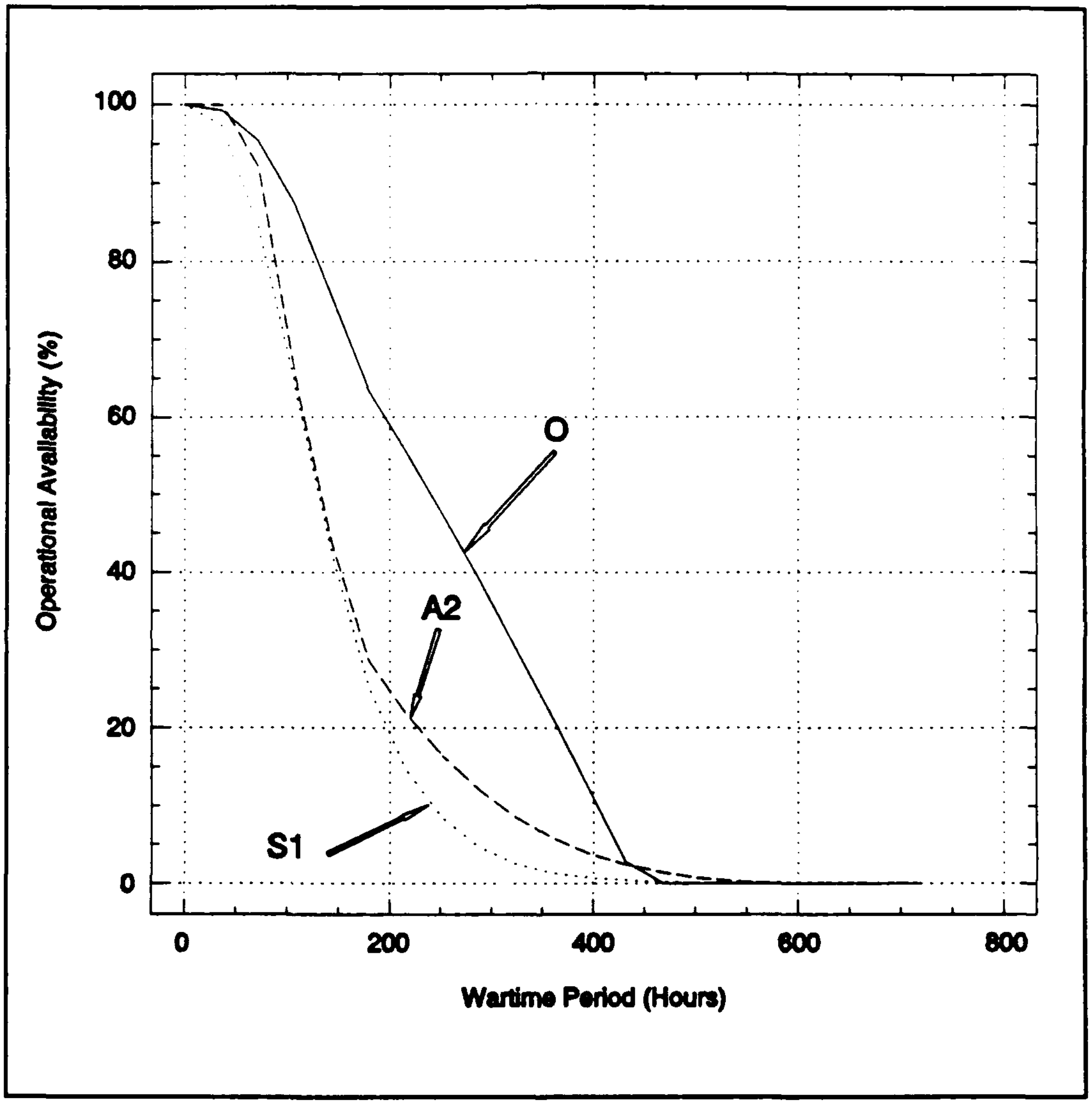

FIGURE 6-18: RESULTS OF OPERATIONAL AVAILABILITY (Ao(t)) FOR MODEL VARIATIONS WITH NO REPAIR CAPABILITY.

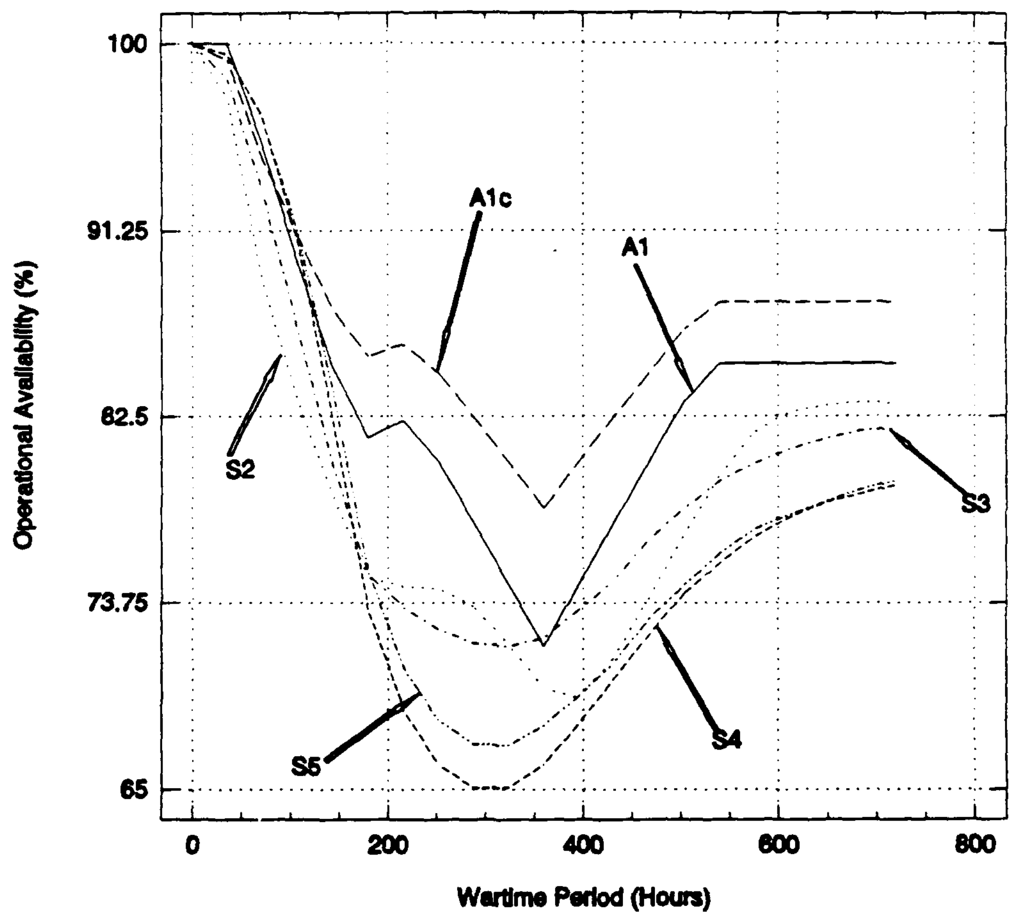

FIGURE 6-19: RESULTS OF OPERATIONAL AVAILABILITY (Ao(t)) FOR MODEL VARIATIONS WITH NO REPAIR CAPABILITY.

4. Analyses on ENMCS(t)

It has been shown in the analytical model that *ENMCS(t)*, whether with or without cannibalization, is related to *Ao(t)* as shown in Equations 3.46 and 3.48, respectively. The same formulae are also appropriate and were used to compute the *ENMCS(t)* in the simulation model. As a consequence, the results shown in Figures 6-20 and 6-21, display "mirror images" of the *Ao(t)* curves.

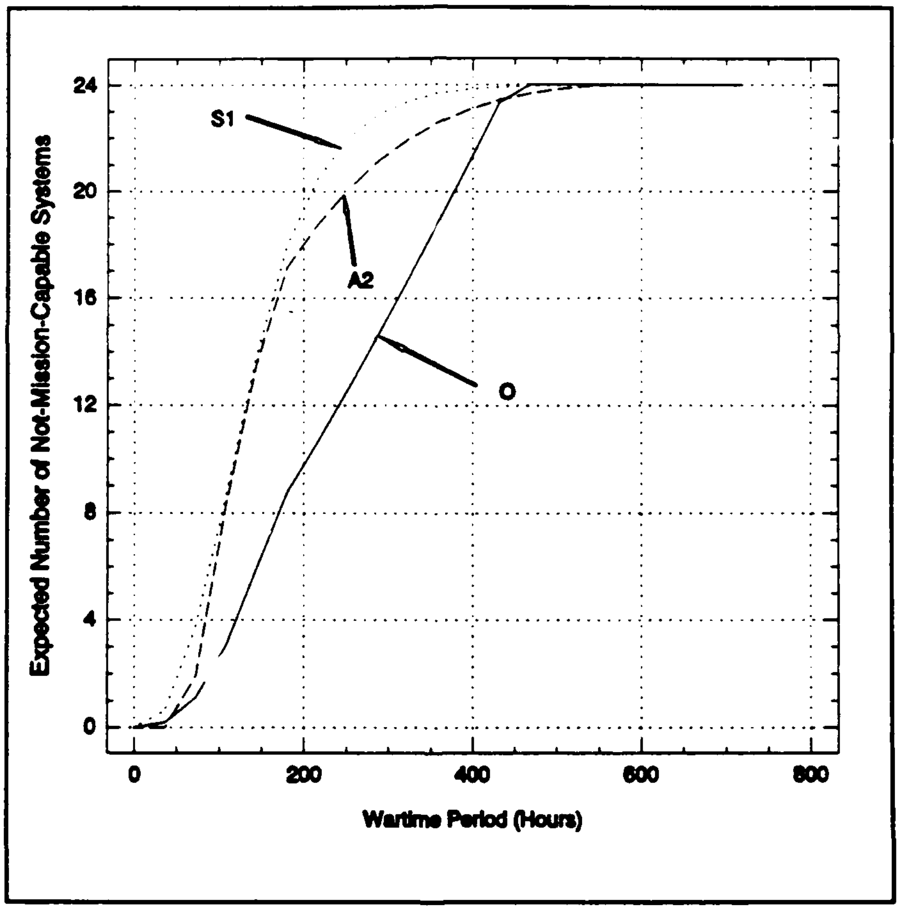

FIGURE 6-20: RESULTS OF EXPECTED NUMBER OF NOT-MISSION-CAPABLE SYSTEMS (ENMCS(t)) FOR MODEL VARIATIONS WITH NO REPAIR CAPABILITY.

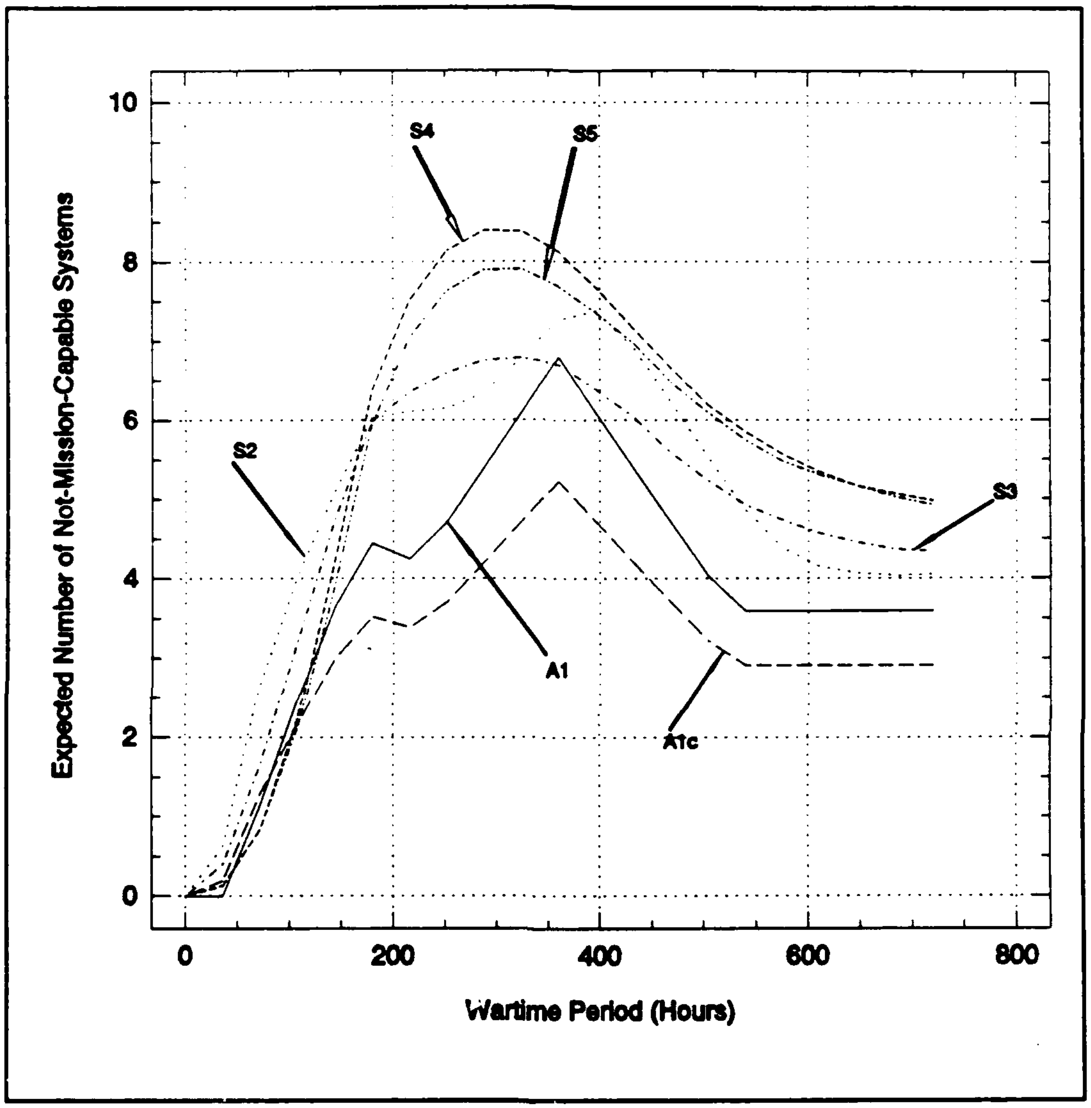

FIGURE 6-21: RESULTS OF EXPECTED NUMBER OF NOT-MISSION-CAPABLE SYSTEMS (ENMCS(t)) FOR MODEL VARIATIONS WITH REPAIR CAPABILITY.

VII. SUMMARY, CONCLUSIONS AND RECOMMENDATIONS

A. SUMMARY

The development of the Wartime Sustainability Model (WSM) for MINDEF began with a discussion of appropriate assumptions, policies and *MOEs*. These provided the framework for the design of both a analytical model and a simulation model for the WSM.

Exact analytical expressions were then derived for the Analytical WSM under several scenarios such as a nonhomogeneous Poisson process for the failure arrivals for the *LRUs* and unlimited repair resources at the repair facilities. The analytical model was then used on a simple example with two-echelon logistics support organization and one-indenture system structure. However, this analytical model is applicable for two-echelon two-indenture problems.

A simulation version of the WSM was also developed to handle variations on repair policies such as limited repair resources and two repair priorities. These variations are extremely difficult to model analytically.

Finally, OPUS-8, a multi-echelon multi-indenture model developed by Systecon AB, was used as an approximation to the analytical version of the WSM for the case of no repair capability. No repair capability is a limitation of the OPUS-8 sustainability option.

When no repair capability and no cannibalization are used in an example, the results of *Total Pipeline, Backorders, Ao(t)* and *ENMCS(t)* are very close for the three versions of WSM. Limited repair capability at the base had the most adverse effect on these measures. Adverse effects were also caused by *LRUs* having high demand rates but insufficient initial inventory levels.

The policy of cannibalization performed better than one without and similarly, LAIF repair priority was shown to improve the *MOEs* slightly when compared to a FCFS policy.

The results also exhibited steady-state characteristics which began at about 450 hours from the start of the war.

B. CONCLUSIONS

Based on the work which was carried out in this thesis, the following conclusions are made:

1. The analytical model is preferred to the simulation model if the underlying assumptions are acceptable. The exact algebraic expressions permit ease of computation and tractability when applied to two-indenture and two-echelon problems. The results from the example indicate that the analytical model and the simulation model for the WSM have comparable outcomes.

2. The simulation model allows one to consider alternative complex repair policies to sustain a military capability under wartime environment. Very few assumptions are needed in its use and therefore it is more suitable for problems for which exact mathematical formulae can not be derived. Unfortunately, this approach is time-consuming especially for multi-indenture and multi-echelon problems.

3. The use of OPUS-8 *in its present form as an approximation to* WSM is only valid when analyzing cases which have no repair capability. However, it is capable of handling a complex military capability comprising different types

of systems, of which each system can have many *LRUs*. This capability is not available in the current WSM.

4. Finally, the numerical results indicate that the *SMOEs* and *MMOEs* are significantly affected by the interactions between spares allocation, repair resource allocation and repair policies such as cannibalization and repair prioritization.

C. RECOMMENDATIONS

The following are recommended for further development:

1. The simple version of the WSM developed in this thesis should be expanded into a general multi-echelon, multi-indenture structure. With a more complicated version, an analytical model may not be possible. For example, the present assumption of independence of all failures should probably be relaxed. In addition, the WSM input parameters need to be varied so that analyses of the results can be more exhaustive. Since the simulation model will undoubtedly be required for such analyses, a more powerful version of MODSIM should be used for this purpose. The present WSM program written in MODSIM provides a suitable basis for such expansion.

2. The ultimate WSM will involve optimization . Such a model should be analytical if possible because exact algebraic formulae can be more readily used in the optimization process. Optimization techniques such as network and dynamic programming should be investigated.

3. OPUS-8 remains the standard spares optimization tool for MINDEF personnel until some version of the WSM is fully implemented. It is recommended that the developer of OPUS-8 should consider the expansion of its current steady-state model into one which can be used to optimize the wartime requirements of spares.

INTENTIONALLY LEFT BLANK

APPENDIX A. MODSIM-II PROGRAM FOR WSM SIMULATION.

{==}
{ WARTIME SUSTAINABILITY MODEL (WSM) SIMULATION WRITTEN IN MODSIM-II BY LIM HUNG HENG FOR THE COMPLETION M.SC (OA) THESIS AT NAVAL POSTGRADUATE SCHOOL }

{== }
The WSM Simulation implemented by MODSIM has the following modules:

MWSM.MOD - WSM is the main module. It declares new instances of objects, controls the other modules and manages replications by checking for Confidence Limit convergence and after each replication, it disposes the appropriate objects to clear computer memory for the next replication.

DGLOBALS.MOD AND
IGLOBALS.MOD- These are the definition and the implementation modules for the *Globals* module respectively (all modules except the main have a definition module and a implementation module). The *Globals* module contains the declaration of all the global variables used by the program and have three global procedures *UTILIZATION, Demand* and *READINPUT*. *UTILIZATION* is ensure the use of the correct utilization rate to compute the *MTBD* and *Demand* is the actual computation of the *MTBD* for any *LRU*. *READINPUT* handles all the input data supplied by the user.

DREPAIR.MOD AND
IREPAIR.MOD - The *Repair* module contains the *Station* object to manage the base repair queuing and servicing. Different policies in repair prioritization will be implemented here.

DEQPT.MOD AND
IEQPT.MOD - The *Eqpt* module contains the generic *Equipment* object to manage all the arrivals of faulty *LRUs* and the decision for base or depot repair. Different policies in repair prioritization will also be implemented here.

DSTOCK.MOD AND
ISTOCK.MOD - The *Stock* module contains the *InvItem* object to manage the recovery of a system either with the use of spare or through cannibalization of previous failed systems.

DREPORT.MOD AND
IREPORT.MOD - The *Report* module collects statistics at fixed interval of time during each replication so that statistical convergence can be managed. The *StatMod* module provided by MODSIM is used heavily to collect the statistics. It can report the current or the final statistics for all the required measures and then dispose of all used memories.

```
{PROGRAM STARTS
{======================================}
{ Definition Module for Globals}
{ Contains data structures which are importable to other modules.
  Objects and variables declared here are essentially the globals
  of the program. }

DEFINITION MODULE Globals;

{ Now import objects and variables from built-in routines and WSM
  specific objects }

FROM IOMod IMPORT StreamObj, FileUseType;

FROM RandMod IMPORT RandomObj;

FROM StatMod IMPORT SINTEGER;

FROM Stock   IMPORT InventoryQueueObj;

FROM GrpMod IMPORT QueueObj;

{ Now declare own objects, procedures and variables }

TYPE

    TimeArray = ARRAY INTEGER OF REAL;

{ ------------------------------------------------------------ }
```

```
    PROCEDURE UTILIZATION(IN Tchange,
                          Trate        : TimeArray;
                          IN NoOfRate  : INTEGER) : REAL;
    { UTILIZATION ensures that the correct utilization is being used
      to compute the demand rate of each LRU }


    PROCEDURE Demand(IN identity: INTEGER;
                     IN rate : REAL;
                     IN system : INTEGER) : REAL;
    { Demand uses the utilization rate, the MTBF, the current
      number of available systems and the QPA to compute the
      mean time between arrivals of the demands }

{ ------------------------------------------------------------- }

    PROCEDURE READINPUT;
    { READINPUT opens all appropriate input and output files and read
      in all the essential input parameters }


{ ------------------------------------------------------------- }

VAR

    IdleStationQueue    : QueueObj;
    { Queue object to track number of idle base repair stations }

    ServiceQueue        : QueueObj;
    { Queue object to enqueue and track the EquipmentObj waiting
      for repair }

    InventoryQueue      : ARRAY INTEGER OF InventoryQueueObj;
    { Queue object to enqueue and track the EquipmentObj waiting
      for spares }

    SystemStat     : ARRAY INTEGER OF SINTEGER;
    { statistical variables to collect weighted statistics
      for the number of available systems at specified time intervals }

    BaseDemandStat : ARRAY INTEGER, INTEGER OF SINTEGER;
    { statistical variables to collect weighted statistics for the
      number of  Total Average Pipeline at specified time intervals }
```

```
InventoryStat  : ARRAY INTEGER, INTEGER OF SINTEGER;
{ statistical variables to collect weighted statistics for the
  number of  Expected Backorders at specified time intervals }

System     : INTEGER;
{ number of available systems at any time }

Inventory  : ARRAY INTEGER OF INTEGER;
{ number of available spares at any time }

SystemDown : ARRAY INTEGER, INTEGER OF INTEGER;
{ number of unavailable systems at any time }

BasePipeNo : ARRAY INTEGER OF SINTEGER;
{ number of total average pipeline at any time }

DepotPipeNo: ARRAY INTEGER OF INTEGER;
{ number of depot pipeline at any time }

InvLevel   : ARRAY INTEGER OF INTEGER;
{ number of initial stock for each LRU }

NoOfItem   : ARRAY INTEGER OF INTEGER;
{ number of LRU in a system }

ItemName   : ARRAY INTEGER OF STRING;
{ name of each LRU }

ItemArrmean : ARRAY INTEGER OF REAL;
{ MTBF of each LRU }

ItemDptmean : ARRAY INTEGER OF REAL;
{ mean depot repair time of each LRU }

ItemBasmean : ARRAY INTEGER OF REAL;
{ mean base repair time of each LRU }

ItemRemtime : ARRAY INTEGER OF REAL;
{ mean removal repair time of each LRU }

ItemReptime : ARRAY INTEGER OF REAL;
{ mean replacement repair time of each LRU }
```

```
    ItemNtrs   : ARRAY INTEGER OF REAL;
    { NRTS of each LRU }

    { input streams }
    InputFile  : StreamObj;
    SystemFile : StreamObj;
    SSystemFile : StreamObj;
    SInventFile : StreamObj;
    SBaseFile   : StreamObj;

    { random streams or seeds }
    BaseVariateStream,
    DepotVariateStream,
    ArrivalVariateStream,
    MTTRVariateStream,
    NTRSVariateStream : RandomObj;

    MaxTime      : REAL;
    SendTime     : REAL;
    ReturnTime   : REAL;
    tolerance    : REAL;

    MaxReplications   : INTEGER;
    CountReplications : INTEGER;
    TimeSample   : INTEGER;
    Tinterval    : INTEGER;
    NoOfRate     : INTEGER;
    NoOfEqpt     : INTEGER;
    NoOfSystem   : INTEGER;
    NoOfStation  : INTEGER;
    NumberDown   : INTEGER;
    MaxCanned    : INTEGER;

    TimeChange,
    Rate         : TimeArray;

END MODULE.
{========================================}

{========================================}
{ Implementation module for Globals }
IMPLEMENTATION MODULE Globals;
```

```
FROM IOMod IMPORT FileUseType(Output,Input);
{ for file handling }

FROM StatMod IMPORT SINTEGER;
{ for statistics collection }

FROM SimMod IMPORT SimTime;

FROM RandMod IMPORT FetchSeed;
{ to obtain random seeds }

FROM Eqpt IMPORT EquipmentObj, IdentityNo;

FROM Report IMPORT ReportObj, CountNoOfNew, CountNoOfDispose;

{--------------------------------------------------------------}
   PROCEDURE UTILIZATION(IN Tchange,
                  Trate      : TimeArray;
                  IN NoOfRate   : INTEGER) : REAL;

   VAR
    util : REAL;
    count : INTEGER;
   BEGIN
    count := 0;
    LOOP
        INC(count);
        IF SimTime() < Tchange[count]
        util := Trate[count];
        RETURN util;
        END IF;
    END LOOP;
   END PROCEDURE;

{--------------------------------------------------------------}
   PROCEDURE Demand(IN identity : INTEGER;
                 IN rate : REAL;
                 IN system : INTEGER) : REAL;

   VAR
    demand : REAL;
   BEGIN
```

```
    demand:= ItemArrmean[identity] * ( 1.0 / ( FLOAT(NoOfItem[identity])
                            * FLOAT(system) * rate));
    RETURN demand;
    END PROCEDURE;

{-------------------------------------------------------------------}

PROCEDURE READINPUT;


VAR

    Report                                   : ReportObj;
    Equipment                                : EquipmentObj;
    seed1, seed2, seed3, seed4, seed5        : INTEGER;
    i, Icount, j, count, NSystem, ILevel, number : INTEGER;
    DemandTime, DemandMean, Urate            : REAL;
    stringdump                               : STRING;

{-------------------------------------------------------------------}

BEGIN

    NEW(SystemFile);
    ASK SystemFile TO Open("SYS.out", Output);

    NEW(Ssystemfile);
    ASK Ssystemfile TO Open("SSYS.out", Output);

    NEW (InputFile);
    ASK InputFile TO Open("wsm.dat", Input);

    ASK InputFile TO ReadInt(seed1);
    ASK InputFile TO ReadLine(stringdump);

    NEW(BaseVariateStream);
    ASK BaseVariateStream TO SetSeed(FetchSeed(seed1));

    ASK InputFile TO ReadInt(seed4);
    ASK InputFile TO ReadLine(stringdump);

    NEW(MTTRVariateStream);
    ASK MTTRVariateStream TO SetSeed(FetchSeed(seed4));
```

```
ASK InputFile TO ReadInt(seed5);
ASK InputFile TO ReadLine(stringdump);

NEW(DepotVariateStream);
ASK DepotVariateStream TO SetSeed(FetchSeed(seed5));

ASK InputFile TO ReadInt(seed2);
ASK InputFile TO ReadLine(stringdump);
NEW(ArrivalVariateStream);
ASK ArrivalVariateStream TO SetSeed(FetchSeed(seed2));

ASK InputFile TO ReadInt(seed3);
ASK InputFile TO ReadLine(stringdump);
NEW(NTRSVariateStream);
ASK NTRSVariateStream TO SetSeed(FetchSeed(seed3));

ASK InputFile TO ReadInt(MaxReplications);
ASK InputFile TO ReadLine(stringdump);
ASK InputFile TO ReadInt(TimeSample);
ASK InputFile TO ReadLine(stringdump);
ASK InputFile TO ReadInt(NoOfStation);
ASK InputFile TO ReadLine(stringdump);
ASK InputFile TO ReadInt(NoOfSystem);
ASK InputFile TO ReadLine(stringdump);
ASK InputFile TO ReadReal(SendTime);
ASK InputFile TO ReadLine(stringdump);
ASK InputFile TO ReadReal(ReturnTime);
ASK InputFile TO ReadLine(stringdump);
ASK InputFile TO ReadInt(NoOfRate);
ASK InputFile TO ReadLine(stringdump);
NEW(TimeChange, 1..NoOfRate);
NEW(Rate, 1..NoOfRate);

ASK InputFile TO ReadLine(stringdump);
count := 0;
WHILE count < NoOfRate
INC(count);

ASK InputFile TO ReadReal(TimeChange[count]);
ASK InputFile TO ReadReal(Rate[count]);
END WHILE;

{ the time for the last rate is the maximum simulation time }
```

```
MaxTime := TimeChange[count];

ASK InputFile TO ReadInt(MaxCanned);
ASK InputFile TO ReadLine(stringdump);
ASK InputFile TO ReadReal(tolerance);
ASK InputFile TO ReadLine(stringdump);
ASK InputFile TO ReadInt(NoOfEqpt);
ASK InputFile TO ReadLine(stringdump);

ASK Sinventfile TO Open("SINV.OUT", Output);
ASK Sbasefile TO Open("SBAS.OUT", Output);

{ Create arrays to tract inventory levels and # in each pipelines }

NEW(NoOfItem, 1..NoOfEqpt);
NEW(ItemName, 1..NoOfEqpt);
NEW(ItemArrmean, 1..NoOfEqpt);
NEW(ItemDptmean, 1..NoOfEqpt);
NEW(ItemBasmean, 1..NoOfEqpt);
NEW(ItemReptime, 1..NoOfEqpt);
NEW(ItemRemtime, 1..NoOfEqpt);
NEW(ItemNtrs, 1..NoOfEqpt);
NEW(InvLevel, 1..NoOfEqpt);
NEW(Inventory, 1..NoOfEqpt);
NEW(BasePipeNo, 1..NoOfEqpt);
NEW(DepotPipeNo, 1..NoOfEqpt);

ASK InputFile TO ReadLine(stringdump);
Icount := 1;
WHILE Icount <= NoOfEqpt

   ASK InputFile TO ReadString(ItemName[Icount]);
   ASK InputFile TO ReadReal(ItemArrmean[Icount]);
   ASK InputFile TO ReadReal(ItemNtrs[Icount]);
   ASK InputFile TO ReadInt(NoOfItem[Icount]);
   ASK InputFile TO ReadInt(InvLevel[Icount]);
   ASK InputFile TO ReadReal(ItemBasmean[Icount]);
   ASK InputFile TO ReadReal(ItemDptmean[Icount]);
   ASK InputFile TO ReadReal(ItemRemtime[Icount]);
   ASK InputFile TO ReadReal(ItemReptime[Icount]);

   INC(Icount);
END WHILE;
```

```
{ Initialize the Global records for cannibalization }

    ASK InputFile TO Close;

  END PROCEDURE;

{---------------------------------------------------------------}
END MODULE.
{=========================================}

{=========================================}

{ Definition Module for Repair }

DEFINITION MODULE Repair;

TYPE

    StationObj = OBJECT

      ASK METHOD ObjInit;
      { method to initialize the StationObj object }

      ASK METHOD FinishRepair;
      { method to ask a station which just finished repair to engage
        the next waiting equipmentObj for service }

    END OBJECT;

END MODULE.
{=========================================}

{=========================================}

{ Implementation module for Repair }
IMPLEMENTATION MODULE Repair;

FROM Eqpt IMPORT EquipmentObj;

FROM Globals IMPORT   IdleStationQueue,
                      ServiceQueue;

OBJECT StationObj;
```

```
{ -------------------------------------------------------------- }
    ASK METHOD ObjInit;

    BEGIN
    END METHOD;

{ -------------------------------------------------------------- }
    ASK METHOD FinishRepair;

    VAR
      Equipment : EquipmentObj;
      numberWaiting : INTEGER;

    BEGIN

      numberWaiting := ASK ServiceQueue numberIn;
      IF numberWaiting = 0
        ASK IdleStationQueue TO Add(SELF);
      ELSE
        Equipment := ASK ServiceQueue TO Remove();
        TELL Equipment TO StartService(SELF);
      END IF;

      END METHOD;

END OBJECT;
END MODULE.
{========================================}

{========================================}

{ Definition module for Stock }
DEFINITION MODULE Stock;

FROM GrpMod IMPORT QueueObj;

TYPE
    InventoryQueueObj = OBJECT(QueueObj)

      ASK METHOD ObjInit;
      { initialize InventoryQueueObj }

      ASK METHOD CheckInventory(IN id : INTEGER) ;
```

```
    { Constantly check for available so that systems can be revived }

  END OBJECT;

  InvItemObj = OBJECT
    Level    : INTEGER;
    id       : INTEGER;
    DownNo   : INTEGER;

    ASK METHOD ObjInit;
    {initialize InvItemObj }

    TELL METHOD Cannibalization;
    { look for downed systems to cannibalize }

    ASK  METHOD TakeInventory;
    { use spare to revive system if available }

  END OBJECT;

VAR

 Downnumber, identNo  : INTEGER;

END MODULE.

{========================================}
{========================================}

{ Implementation module for Stock }
IMPLEMENTATION MODULE   Stock;

FROM Globals IMPORT     Inventory,
                        InventoryQueue,
                        System,
                        ItemReptime,
                        MaxCanned,
                        SystemDown;


{-------------------------------------------}
OBJECT InventoryQueueObj;
```

```
    ASK METHOD ObjInit;
    BEGIN
    END METHOD;

{------------------------------------------------------------}

    ASK METHOD CheckInventory(IN id : INTEGER) ;

    VAR
       InvItem  : InvItemObj;
       numberWaiting : INTEGER;

    BEGIN

    IF Inventory[id] > 0

       IF (ASK  SELF numberIn) > 0

          InvItem := ASK SELF TO Remove();
          ASK InvItem TO TakeInventory;

       END IF;

    END IF;

       END METHOD;

END OBJECT;

{------------------------------------------------------------}

OBJECT InvItemObj;

    ASK METHOD ObjInit;
    BEGIN

     DownNo := Downnumber;
     id     := identNo;

    END METHOD;

{------------------------------------------------------------}
```

```
 TELL METHOD Cannibalization;
{ use existing inventory to recover system as fast as possible.
 only when there is no spares left, then start to cannibalize.
 However cannibalization takes some time, use constant }

 VAR

  Down, numberzero : INTEGER;

 BEGIN

  Level := Inventory[id];

  IF Level > 0

{ since there is spare, can use it to revive system }

{ time to replace faulty part }
    WAIT DURATION ItemReptime[id]
      ON INTERRUPT
    END WAIT;

    ASK  SELF TO TakeInventory;

{-----------------------------------------------------------------}
  ELSE
{ No spares available Cannibalization Policy START }

  IF DownNo > MaxCanned

{ Since cannibalization not allowed, must wait for spare to be available }

     ASK InventoryQueue[id] TO Add(SELF);
     TERMINATE;
  ELSE

{ start cannibalizing on the downed equipment }

    Down := 1;
    IF DownNo > 1
     numberzero := 0;

     WHILE Down <= DownNo - 1
```

```
      IF SystemDown[Down, id] > 0

        SystemDown[Down, id] := SystemDown[Down, id] - 1;

{ Since able to cannibalize, let the current downed system be up after
  a constant time }
{
        WAIT DURATION 3.0
          ON INTERRUPT
        END WAIT;
}
{ downed System is now available after cannibalization. }

        INC(System);
        DISPOSE(SELF);

{ get out of while loop }
        TERMINATE;

      ELSE

{ Count no of downed systems which cannot be cannibalization }
        numberzero := numberzero + 1;

      END IF;

      Down := Down + 1;
    END WHILE;

    IF numberzero = DownNo - 1

{ Let one more system be unavailable for cannibalization }
          SystemDown[DownNo, id] :=
                  SystemDown[DownNo, id] - 1;

        DISPOSE(SELF);
        TERMINATE;
    END IF;


  ELSIF DownNo = 1

      IF SystemDown[1, id] > 0
```

```
        SystemDown[1, id] := SystemDown[1, id] - 1;
{ Since able to cannibalize, let the current downed system be up after
 a constant time }
{
        WAIT DURATION 3.0
         ON INTERRUPT
        END WAIT;
}
{ downed System is now available after cannibalization }
        INC(System);

        DISPOSE(SELF);
        TERMINATE;
       ELSE

     ASK InventoryQueue[id] TO Add(SELF);

        TERMINATE;
       END IF;

    END IF;

  END IF;
 END IF;

END METHOD;

{------------------------------------------------------------}
     ASK METHOD TakeInventory;

     BEGIN
{ take inventory now }
       DEC(Inventory[id]);
{ now can make system available }
       INC(System);

       ASK InventoryQueue[id] TO CheckInventory(id);

       DISPOSE(SELF);

     END METHOD;

END OBJECT;
```

```
END MODULE.
{=========================================}

{=========================================}

{ Definition Module for Eqpt }
DEFINITION MODULE Eqpt;

FROM Repair IMPORT StationObj;

TYPE
   EquipmentObj = OBJECT
      Station   : StationObj;
      identity  : INTEGER;
      startTime : REAL;

      ASK METHOD ObjInit;
      { method to initialize the EquipmentObj object }

      TELL METHOD EnterSystem;
      { method to sample for a new equipment to fail and then to
        direct the current failed equipment for appropriate repair
        actions }

      TELL METHOD CheckSystem;
      { method to direct the failed equipment to check for available
        spare or to cannibalize a good part from downed part. To do
        this a new object InvItem is declared for this purpose. InvItem
        has its own methods for cannibalization. EnterSystem also tell
        the equipment to Chooserepair simultaneous }

      TELL METHOD ChooseRepair;
      { ChooseRepair decides by uniform random sampling whether an
        equipment goes to base or depot for repair.
        For base repair, it will check for available base repair
        station and then tell it to StartService }

      TELL METHOD StartService(IN Station: StationObj);
      { StartService samples a repair time to complete the repair
        of a failed equipment }

   END OBJECT;
```

```
VAR
    IdentityNo : INTEGER;

END MODULE.
{======================================}

{======================================}

{ Implementation module for Eqpt }
IMPLEMENTATION MODULE Eqpt;

FROM SimMod IMPORT SimTime;
{ SimTime shows the current simulation time }

FROM UtilMod IMPORT ExitToOS;
{ ExitToOS terminates the program when a condition is satisfied }

FROM Repair IMPORT StationObj;

FROM Report IMPORT CountNoOfNew, CountNoOfDispose;

FROM Stock  IMPORT InvItemObj, identNo, Downnumber;

FROM Globals IMPORT   InputFile,
                      DepotVariateStream,
                      ArrivalVariateStream,
                      NTRSVariateStream,
                      MTTRVariateStream,
                      BaseVariateStream,
                      MaxTime,
                      SendTime,
                      ReturnTime,
                      TimeChange,
                      Rate,
                      NoOfRate,
                      NoOfEqpt,
                      NoOfItem,
                      ItemName,
                      ItemArrmean,
                      ItemDptmean,
                      ItemBasmean,
                      ItemRemtime,
                      ItemReptime,
```

```
                ItemNtrs,
                NoOfSystem,
                System,
                NoOfStation,
                SystemDown,
                MaxCanned,
                Inventory,
                InvLevel,
                BasePipeNo,
                DepotPipeNo,
                ServiceQueue,
                IdleStationQueue,
                InventoryQueue,
                UTILIZATION,
                Demand;

{----------------------------------------------------}
OBJECT EquipmentObj;

{----------------------------------------------------}
 ASK METHOD ObjInit;

   BEGIN
      identity    := IdentityNo;
 END METHOD;

{----------------------------------------------------}
 TELL METHOD EnterSystem;

     VAR
      Equipment2: EquipmentObj;
      interarrivalTime : REAL;
      DemandMean : REAL;
      Urate : REAL;
      nsys, error  : INTEGER;
{----------------------------------------------------}

 BEGIN

  startTime := SimTime();

{ Start another object of the same type }
```

```
        Urate := UTILIZATION(TimeChange, Rate, NoOfRate);
        nsys := System;

{ get out if divide by 0 }

        IF nsys = 0
        ExitToOS(error);
        END IF;

    DemandMean := Demand(identity, Urate, nsys);
    interarrivalTime := ASK ArrivalVariateStream Exponential(DemandMean);
    IF ( SimTime() + interarrivalTime ) <   MaxTime

      IdentityNo        :=   identity  ;

      NEW(Equipment2);

      INC(CountNoOfNew);

 { Schedule the next equipment of the same type to arrive }

        TELL Equipment2 TO EnterSystem IN interarrivalTime;

     END IF;

 TELL SELF TO CheckSystem;

  END METHOD;

{------------------------------------------------------------}

  TELL METHOD CheckSystem;

  VAR
        Station : StationObj;
        InvItem : InvItemObj;
        I, numberIdle, nsys, ninv,
        numberzero, Down          : INTEGER;
        depottime  : REAL;
        basetime   : REAL;
        ratio      : REAL;
        mttremove  : REAL;
```

```
  BEGIN

{ add one to the Basepipeline }

 INC(BasePipeNo[identity]);

{----------------------------------------------------------------}
{ wait for diagnosis and removal of failed part in by exponential assumption }

 mttremove := ASK MTTRVariateStream  Exponential(ItemRemtime[identity]);
 WAIT DURATION mttremove


{ downtime experience by a System }
   DEC(System);
{ Check the current no of downed systems }
   Downnumber := NoOfSystem - System;

 END WAIT;
{----------------------------------------------------------------}

{ cannibalization policy to revive system }
 identNo := identity;
 NEW(InvItem);
 TELL InvItem TO Cannibalization;

{ send the faulty item for repair now }
 TELL SELF TO ChooseRepair;

 END METHOD;

{----------------------------------------------------------------}
  TELL METHOD ChooseRepair;

  VAR
      Station : StationObj;
      I, numberIdle, nsys, ninv  : INTEGER;
      depottime  : REAL;
      basetime   : REAL;
      ratio      : REAL;
      mttremove  : REAL;

  BEGIN
```

```
{ decision point for base/depot repair }

   ratio := ASK NTRSVariateStream  UniformReal( 0.0, 1.0 );

      IF ratio > ItemNtrs[identity]

{ Base Repair starts }

         numberIdle := ASK IdleStationQueue numberIn;

         IF numberIdle = 0
{ Enqueue the customer }
            ASK ServiceQueue TO Add(SELF);

         ELSE
{ The customer will start service immediately }
            Station := ASK IdleStationQueue TO Remove();

            TELL SELF TO StartService(Station);
         END IF;

    ELSE

{ transportation time constant }
        WAIT DURATION SendTime;
         ON INTERRUPT
        END WAIT;

{ add one to depotpipeline }

  INC(DepotPipeNo[identity]);

{ repairtime depends on ItemDptmean[identity] }
        depottime := ASK DepotVariateStream
                  Exponential(ItemDptmean[identity]);
        WAIT DURATION depottime;
         ON INTERRUPT
        END WAIT;

{ minus one to depot pipeline }

  DEC(DepotPipeNo[identity]);
```

```
{ transportation time constant }
      WAIT DURATION ReturnTime;
        ON INTERRUPT
      END WAIT;

{ basepipeline minus one since the part is sent to depot for repair
  completed repair }

  DEC(BasePipeNo[identity]);

{ downed item is now available for Inventory[identity] }

      INC(Inventory[identity]);

{ dispose the equipment to clear memory }
    DISPOSE(SELF);

    INC(CountNoOfDispose);

  END IF;

  END METHOD;

{----------------------------------------------------------------}

    TELL METHOD StartService(IN Station: StationObj);

    VAR
      ServiceTime : REAL;

    BEGIN

      ServiceTime := ASK BaseVariateStream
                       Exponential(ItemBasmean[identity]);
      WAIT DURATION ServiceTime
        ON INTERRUPT
      END WAIT;

      ASK Station TO FinishRepair;

{ downed item is now available for Inventory[identity] }

    INC(Inventory[identity]);
```

```
{ minus one to the Basepipeline }

       DEC(BasePipeNo[identity]);

     DISPOSE(SELF);
     INC(CountNoOfDispose);
     END METHOD;

{------------------------------------------------------------------}
END OBJECT;

END MODULE.
{ = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = }

{ = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = }

{ Definition module for Report }

DEFINITION MODULE Report;

TYPE

     ReportObj = OBJECT

        ASK METHOD ObjInit;
        { initialize ReportObj }

        ASK METHOD ReportNow;
        { asking for statistical collections at specified intervals }

        ASK METHOD FinalReport;
        { asking  for final statistical results with confidence intervals }

        TELL METHOD ReportStatusNow;
        { asking for statistical collections at specified intervals }

     END OBJECT;

VAR
  CountNoOfNew : INTEGER;
  CountNoOfDispose : INTEGER;
```

```
END MODULE.
{========================================}

{========================================}

{ Implementation module for Report }
IMPLEMENTATION MODULE Report;

FROM Globals IMPORT  MaxTime, System, Inventory, InvLevel,
                     SystemFile,
                     Sinventfile, Ssystemfile, Sbasefile,
                     Tinterval, TimeSample,
                     SystemDown, NoOfSystem, MaxCanned,
                     BasePipeNo, DepotPipeNo,
                     CountReplications,
                     ItemName,
                     SystemStat,
                     BaseDemandStat,
                     InventoryStat,
                     NoOfEqpt;

FROM SimMod IMPORT  SimTime;

FROM MathMod IMPORT  SQRT;

FROM StatMod IMPORT  IStatObj;

OBJECT ReportObj;

{ -------------------------------------------------------- }
   ASK METHOD ObjInit;

   VAR
    i, j :  INTEGER;

   BEGIN

{ create new statistical variables to collect weighted statistics }

     NEW(SystemStat, 1..TimeSample+1);
     NEW(BaseDemandStat, 1..TimeSample+1, 1..NoOfEqpt);
     NEW(InventoryStat, 1..TimeSample+1, 1..NoOfEqpt);
```

```
    FOR i := 1 TO TimeSample+1

      ASK (GETMONITOR(SystemStat[i],Istatobj))
         TO SetHistogram(0,NoOfSystem,1);

      FOR j := 1 TO NoOfEqpt

         ASK (GETMONITOR(BaseDemandStat[i,j],Istatobj))
             TO SetHistogram(0,100,1);
         ASK (GETMONITOR(InventoryStat[i,j],Istatobj))
             TO SetHistogram(0, InvLevel[j] ,1);

      END FOR;
    END FOR;
  END METHOD;

{ -------------------------------------------------------------- }
  ASK METHOD ReportNow;

 VAR
  identity        : INTEGER;
 BEGIN

      ASK SystemFile TO WriteReal(SimTime(), 9,2);
      ASK SystemFile TO WriteInt(System, 6);
      ASK SystemFile TO WriteLn;

      SystemStat[Tinterval] := System;

 FOR identity := 1 TO NoOfEqpt


   IF BasePipeNo[identity] - InvLevel[identity] >  0

     InventoryStat[Tinterval,identity]  :=
           BasePipeNo[identity] - InvLevel[identity];
   ELSE
     InventoryStat[Tinterval,identity]  := 0;
   END IF;

   BaseDemandStat[Tinterval,identity]  := BasePipeNo[identity] ;

 END FOR;
```

```
    TELL SELF TO ReportStatusNow;

   END METHOD;

{ ------------------------------------------------------------------- }

  TELL METHOD ReportStatusNow;

  BEGIN

  IF SimTime() < MaxTime

{ wait a predetermined interval to call report again }

     WAIT DURATION (MaxTime/FLOAT(TimeSample));
     END WAIT;
     Tinterval := Tinterval + 1;

     ASK SELF TO ReportNow;

  END IF;

 END METHOD;

{ ------------------------------------------------------------------- }

   ASK METHOD FinalReport;

  VAR
   i, j          : INTEGER;
   sysmean, syslowerlimit, sysupperlimit, ssdev, tolerance : REAL;
   basemean, baselowerlimit, baseupperlimit, bsdev, tol    : REAL;
   invmean, invlowerlimit, invupperlimit, isdev           : REAL;

  BEGIN
ASK Ssystemfile TO WriteString("FINAL STATISTICS FOR SYSTEM
AVAILABILITY,
                        REPLICATIONS = ");

ASK Ssystemfile TO WriteInt(CountReplications, 5);
ASK Ssystemfile TO WriteLn;
ASK Ssystemfile TO WriteString(" Sim.Time SysMean SysLLimit SysULimit");
ASK Ssystemfile TO WriteLn;
```

```
ASK Sbasefile TO WriteString("FINAL STATISTICS FOR BASE DEMAND
NUMBERS,
                         REPLICATIONS = ");
ASK Sbasefile TO WriteInt(CountReplications, 5);
ASK Sbasefile TO WriteLn;

  ASK Sbasefile TO WriteString(" Sim.Time");
FOR j := 1 TO NoOfEqpt
  ASK Sbasefile TO WriteString(" ITEM  =     ");
  ASK Sbasefile TO WriteString(ItemName[j]);
  ASK Sbasefile TO WriteString("          ");
END FOR;
ASK Sbasefile TO WriteLn;

  ASK Sbasefile TO WriteString("         ");
FOR j := 1 TO NoOfEqpt
  ASK Sbasefile TO WriteString("    Mean    LLimit   ULimit    |");
END FOR;
ASK Sbasefile TO WriteLn;
ASK Sbasefile TO WriteLn;

ASK Sinventfile TO WriteString("FINAL STATISTICS FOR INVENTORY
LEVELS,
                          REPLICATIONS = ");
ASK Sinventfile TO WriteInt(CountReplications, 5);
ASK Sinventfile TO WriteLn;

  ASK Sinventfile TO WriteString(" Sim.Time");
FOR j := 1 TO NoOfEqpt
  ASK Sinventfile TO WriteString(" ITEM  =     ");
  ASK Sinventfile TO WriteString(ItemName[j]);
  ASK Sinventfile TO WriteString("          ");
END FOR;
ASK Sinventfile TO WriteLn;

  ASK Sinventfile TO WriteString("         ");
FOR j := 1 TO NoOfEqpt
  ASK Sinventfile TO WriteString("    Mean    Llimit   Ulimit    |");
END FOR;
ASK Sinventfile TO WriteLn;
ASK Sinventfile TO WriteLn;

      FOR i := 1 TO TimeSample + 1
```

```
ASK Ssystemfile TO WriteReal( FLOAT(i-1)
          * (MaxTime/FLOAT(TimeSample)), 9,2);

ASK Sbasefile TO WriteReal( FLOAT(i-1)
          * (MaxTime/FLOAT(TimeSample)), 9,2);

ASK Sinventfile TO WriteReal( FLOAT(i-1)
          * (MaxTime/FLOAT(TimeSample)), 9,2);

sysmean  := GETMONITOR(SystemStat[i],Istatobj).Mean();
ssdev    := GETMONITOR(SystemStat[i],Istatobj).StdDev();

syslowerlimit := sysmean - (2.0 *
            ssdev/SQRT(FLOAT(CountReplications)));
sysupperlimit := sysmean + (2.0 *
            ssdev/SQRT(FLOAT(CountReplications)));

ASK Ssystemfile TO WriteReal(sysmean, 9,3);
ASK Ssystemfile TO WriteReal(syslowerlimit, 9,3);
ASK Ssystemfile TO WriteReal(sysupperlimit, 9,3);
ASK Ssystemfile TO WriteLn;

FOR j := 1 TO NoOfEqpt

  basemean := GETMONITOR(BaseDemandStat[i,j],Istatobj).Mean();
  bsdev    := GETMONITOR(BaseDemandStat[i,j],Istatobj).StdDev();
  baselowerlimit := basemean - (2.0 *
             bsdev/SQRT(FLOAT(CountReplications)));
  baseupperlimit := basemean + (2.0 *
             bsdev/SQRT(FLOAT(CountReplications)));

  ASK Sbasefile TO WriteReal(basemean, 9,3);
  ASK Sbasefile TO WriteReal(baselowerlimit, 9,3);
  ASK Sbasefile TO WriteReal(baseupperlimit, 9,3);
  ASK Sbasefile TO WriteString("   |");

  invmean := GETMONITOR(InventoryStat[i,j],Istatobj).Mean();
  isdev    := GETMONITOR(InventoryStat[i,j],Istatobj).StdDev();
  invlowerlimit := invmean - (2.0 *
             isdev/SQRT(FLOAT(CountReplications)));
  invupperlimit := invmean + (2.0 *
             isdev/SQRT(FLOAT(CountReplications)));
```

```
        ASK Sinventfile TO WriteReal(invmean, 9,3);
        ASK Sinventfile TO WriteReal(invlowerlimit, 9,3);
        ASK Sinventfile TO WriteReal(invupperlimit, 9,3);
        ASK Sinventfile TO WriteString("   |");

      END FOR;

      ASK Sbasefile TO WriteLn;
      ASK Sinventfile TO WriteLn;

    END FOR;

  END METHOD;

{ ------------------------------------------------------------------ }
END OBJECT;
END MODULE.
{=========================================================}

{=========================================================}
{ Main Module for the WSM Simulation }
MAIN MODULE WSM;

FROM SimMod IMPORT StartSimulation, SimTime, ResetSimTime;

FROM IOMod IMPORT FileUseType(Output,Input);

FROM StatMod IMPORT Istatobj;

FROM MathMod IMPORT SQRT;

FROM Eqpt IMPORT EquipmentObj, IdentityNo;

FROM Report IMPORT ReportObj, CountNoOfNew, CountNoOfDispose;

FROM Stock IMPORT InvItemObj;

FROM Repair IMPORT StationObj;

FROM Globals IMPORT
            Ssystemfile,

            MTTRVariateStream,
```

BaseVariateStream,
DepotVariateStream,

ArrivalVariateStream,
NTRSVariateStream,

MaxReplications,
CountReplications,
MaxTime,
TimeSample,
Tinterval,
SendTime,
ReturnTime,
TimeChange,
Rate,
NoOfRate,
System,
SystemDown,
NoOfEqpt,
NoOfItem,
ItemName,
ItemArrmean,
ItemDptmean,
ItemBasmean,
ItemReptime,
ItemRemtime,
ItemNtrs,
NoOfSystem,
MaxCanned,
tolerance,
Inventory,
NoOfStation,
InvLevel,
BasePipeNo,
DepotPipeNo,
InventoryQueue,
IdleStationQueue,
ServiceQueue,
SystemStat,
BaseDemandStat,
InventoryStat,
UTILIZATION,
READINPUT,

```
                    Demand;

VAR

    Station                              : StationObj;
    InvItem                               : InvItemObj;
    Report                                : ReportObj;
    Equipment                              : EquipmentObj;
    i, Icount, j, count, Nsystem, Ilevel, number : INTEGER;
    DemandTime, DemandMean, Urate                : REAL;
    sysmean, ssdev          : REAL;
    basemean,  bsdev, tol     : REAL;
    invmean,   isdev        : REAL;
    countconvergence          : INTEGER;

{ ------------------------------------------------------------- }

BEGIN

{ procedure to read in datafile }
 READINPUT;

    NEW(InventoryQueue, 1..NoOfEqpt);

    NEW(SystemDown, 1..MaxCanned, 1..NoOfEqpt);
    NEW(Report);


{ do replications for statistics collection }

  CountReplications := 1;

{ ------------------------------------------------------------- }
{ loop for replications until statistical convergence is reached }
  LOOP

   ResetSimTime(0.0);
   Tinterval := 1;

   System := (NoOfSystem);

   NEW(ServiceQueue);
   NEW(IdleStationQueue);
```

```
   FOR i := 1 TO NoOfStation
      NEW(Station);
      ASK IdleStationQueue TO Add(Station);
   END FOR;

   FOR Icount := 1 TO NoOfEqpt

     IdentityNo := Icount;
     NEW(Equipment);

     INC(CountNoOfNew);

{ initialize the pipelines }

     BasePipeNo[Icount]  := 0;
     DepotPipeNo[Icount] := 0;

     NEW(InventoryQueue[Icount]);
     Inventory[Icount] := (InvLevel[Icount]);

     Urate := UTILIZATION(TimeChange, Rate, NoOfRate);
     DemandMean := Demand(Icount, Urate, NoOfSystem);
     DemandTime := ASK ArrivalVariateStream Exponential(DemandMean);

     TELL Equipment TO EnterSystem IN DemandTime;
   END FOR;

{ need to initialize the NoOfItem[Icount] for each item }

     FOR i := 1 TO MaxCanned
        FOR j := 1 TO NoOfEqpt
           SystemDown[i, j] := NoOfItem[j];
        END FOR;
     END FOR;

     ASK Report TO ReportNow;

{ ask MODSIM to start simulation }
     StartSimulation;

{ dispose of all remaining objects to clear memory }
   number := ASK IdleStationQueue numberIn;
   IF number <> 0
```

```
    FOR i := 1 TO number
      Station := ASK IdleStationQueue TO Remove();
      DISPOSE(Station);
    END FOR;
  END IF;
  DISPOSE(IdleStationQueue);

   number := ASK ServiceQueue numberIn;
   IF number <> 0
    FOR i := 1 TO number
      Equipment := ASK ServiceQueue TO Remove();
      DISPOSE(Equipment);
      INC(CountNoOfDispose);
    END FOR;
  END IF;
  DISPOSE(ServiceQueue);

{ Dispose InvItem that are not processed }
    FOR i := 1 TO NoOfEqpt
      number := ASK InventoryQueue[i] numberIn;

      IF number <> 0
        FOR j := 1 TO number
          InvItem   := ASK InventoryQueue[i] TO Remove();
          IF InvItem <> NILOBJ
            DISPOSE(InvItem);
          END IF;
        END FOR;
      END IF;
      DISPOSE(InventoryQueue[i]);
    END FOR;

{
    ASK Ssystemfile TO WriteLn;
    ASK Ssystemfile TO WriteString("# NEW = ");
    ASK Ssystemfile TO WriteInt(CountNoOfNew, 5);
    ASK Ssystemfile TO WriteLn;
    ASK Ssystemfile TO WriteString("# DISPOSED = ");
    ASK Ssystemfile TO WriteInt(CountNoOfDispose, 5);
    ASK Ssystemfile TO WriteLn;
}
    OUTPUT("REPLICATION NUMBER = ", CountReplications);
```

```
{ check for statistical convergence }

 IF CountReplications > 5
  countconvergence := 0;
  FOR i := 1 TO TimeSample + 1

   sysmean := GETMONITOR(SystemStat[i],Istatobj).Mean();
   ssdev   := GETMONITOR(SystemStat[i],Istatobj).StdDev();

   tol     := (2.0 * ssdev/SQRT(FLOAT(CountReplications)));
   IF (tol <= sysmean * tolerance )
      INC(countconvergence)
   END IF;

      FOR j := 1 TO NoOfEqpt

       basemean := GETMONITOR(BaseDemandStat[i,j],Istatobj).Mean();
       bsdev    := GETMONITOR(BaseDemandStat[i,j],Istatobj).StdDev();
       tol      := (2.0 * bsdev/SQRT(FLOAT(CountReplications)));

      IF (tol <= basemean * tolerance )
        INC(countconvergence)
      END IF;

       invmean := GETMONITOR(InventoryStat[i,j],Istatobj).Mean();
       isdev   := GETMONITOR(InventoryStat[i,j],Istatobj).StdDev();
       tol     := (2.0 * isdev/SQRT(FLOAT(CountReplications)));

      IF (tol <= invmean * tolerance )
        INC(countconvergence)
      END IF;

      END FOR;

   END FOR;

   OUTPUT("CONVERGENCE NUMBER = ", countconvergence);

   IF countconvergence >= (TimeSample+1 +
                  ((TimeSample+1)*NoOfEqpt*2) )
     EXIT
   END IF;
 END IF;
```

```
    IF  CountReplications > MaxReplications
       EXIT
    END IF;

  INC(CountReplications);

{ replications end }
  END LOOP;

{ Report Final Statistics Now }

ASK Report TO FinalReport;

    OUTPUT("ENDED NORMALLY");

END MODULE.
{===========================================}
{ END OF MODSIM PROGRAM }
```

APPENDIX B. MATHLAB PROGRAM FOR ANALYTICAL WSM.

```
{=======================================}
%*********************************************************************
%ANALYTICAL WAR SUSTAINABILITY MODEL (WSM) PROGRAM
%  WRITTEN IN PC-MATLAB BY LIM HUNG HENG FOR THE
%            COMPLETION OF M.SC (OA) THESIS
%             AT NAVAL POSTGRADUATE SCHOOL
%*********************************************************************
%This MATLAB program used the assumptions and analytical solutions based
%on Chapter III - Analytical Model Development and the wartime scenario
%given in Chapter V - A Case Study.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%DATA INPUT PARAMETERS:
%--------------------------------------------------------------------
% itemfr      = Failure rate (failures / 1,000,000 hours) of each LRU
% itemnrts    = The proportion of repair not repairable at the base
%               (nrts =  Not-Repairable This Station) for each LRU.
% itemqpa     = The quantity of each LRU found in each system.
%               (qpa  =  quantity per application).
% invlevel    = The initial inventory level stocked at the base for each LRU.
% brtime      = Base repair time for each LRU.
% dptime        = Depot repair time for each LRU.
% totalsys    = Total number of available systems at the start of war period.
% nmcs        = Maximum number of Not-Mission Capable Systems to be
%               tolerated by the operational users (an operational target).
% fortime     = Transportation time from base to depot.
% rettime     = Transportation time from depot to base.
% timchg      = time at which utilization rate of the system changes.
% urate       = different levels of utilization rates of the system
%               (to be determined by operational users).
% notimint    = number of time intervals to be analyzed.
%--------------------------------------------------------------------

itemfr    = [390 4040 360 11510 4490 6070 570 4480 17000 10920];
itemnrts  = [.07 .06 .21 .59 .04 .04 .24 .09 .06 .28];
itemqpa   = [1 1 1 1 1 1 1 1 1 1];
invlevel  = [1 3 1 5 2 4 1 2 6 2];
brtime    = [48 48 72 72 48 48 72 48 48 48];
dptime    = [120 120 120 120 120 120 120 120 120 120];
```

```
totalsys   = 24;
nmcs       =  4;
fortime    = 120;
rettime    = 120;
timchg     = [168 720];
urate      = [0.2 0.1];
notimint   = 100;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Calculate equi-distance length between time intervals.
timeint = 0:(timchg(2)/notimint):timchg(2);
%-----------------------------------------------------------------

%make itemnrts into matrix with time intervals as rows and LRUs as columns.
itemnrts = ones(timeint)' * itemnrts;
%-----------------------------------------------------------------

%Calculate the demand rate for each LRU at each time intervals.
%based on the given utilization rates.
demrat1 = (1.e-6 * totalsys * urate(1)) * (itemfr);
demrat1 = ones(timeint)' * demrat1;
demrat2 = (1.e-6 * totalsys * urate(2)) * (itemfr);
demrat2 = ones(timeint)' * demrat2;
%-----------------------------------------------------------------

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%base repair service time for each LRU at each time intervals.
basetime = ones(timeint)' * brtime;

%timedim is just a matrix of time intervals used for this computation.
timedim = timeint' * ones(brtime);

%-----------------------------------------------------------------
%compute the various base service times at which the LRU is still held up.

%service time before reaching the basetime.
%bastim11 depends only on timedim.
i = (timedim <= basetime);
bastim11= (i) .* timedim;

%-----------------------------------------------------------------
%service time after the basetime but before the change from high
%utilization to low utilization.
```

```
%bastim12 has a constant time.
i = (timedim > basetime) & (timedim <= timchg(1));
bastim12 = (i) .* basetime;

%----------------------------------------------------------------
%service time during low utilization but before (timchg(1)+basetime)
%since we know that LRUs that failed under the high-utilization demand
%time timchg(1) must wait for the whole basetime to be repaired.

i = (timedim > timchg(1)) & (timedim <=  (timchg(1) + basetime) );
bastim13 = (i) .* ((timchg(1) + basetime) - timedim);
bastim21 = (i) .* (timedim - timchg(1));

%service time after (basetime+timchg(1))
%bastim22 depends only on basetime.
i = (timedim > (timchg(1) + basetime));
bastim22 = (i) .* basetime;

%----------------------------------------------------------------
%bastim1 is the matrix addition to be used together with demrat1 the
%high-utilization demand rate
%bastim2 is the matrix addition to be used together with demrat2 the
%low-utilization demand rate
bastim1 = bastim11 + bastim12 + bastim13;
bastim2 = bastim21 + bastim22;
basdemand1 = ((1 - itemnrts) .* demrat1) .* bastim1;
basdemand2 = ((1 - itemnrts) .* demrat2) .* bastim2;

%----------------------------------------------------------------
%basdemand is the mean demand caused by the depot turnaround time.
basdemand  = basdemand1 + basdemand2;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%depot repair service time for each LRU at each time intervals.

%since dptime is constant, the depot turnaround time from base to
%depot is given below.
depotime = fortime + rettime + dptime;
depotime = ones(timeint)' * depotime;
%compute the possible depot demand for high utilization

%----------------------------------------------------------------
%compute the various depot service times at which the LRU is still held up.
```

```
%service time before the change from high utilization to low utilization.
%dptim11 depends only on timdim.
i = (timedim <= timchg(1));
dpttim11= (i) .* timedim;

%-----------------------------------------------------------------
%service time during low utilization but before the depot turnaround time.
%dptim12 has a constant time and dpttim21 depends on (timedim - timchg(1)).
i = (timedim > timchg(1)) & (timedim <= depotime );
dpttim12 = (i) .* timchg(1);
dpttim21 = (i) .* (timedim - timchg(1));

%-----------------------------------------------------------------
%service time between the depot turnaround time and (depotime+timchg(1))
%since we know that LRUs that failed under the high-utilization demand
%time timchg(1) must wait for the whole depottime to be repaired.
i = (timedim > depotime) & (timedim <= (timchg(1) + depotime) );
dpttim13 = (i) .* ((timchg(1)+depotime) - timedim);
dpttim22 = (i) .* (timedim - timchg(1));

%-----------------------------------------------------------------
%service time after (depotime+timchg(1))
%dptim23 depends only on depotime.
i = (timedim > (timchg(1) + depotime) );
dpttim23 = (i) .* depotime;

%-----------------------------------------------------------------
%dptim1 is the matrix addition to be used together with demrat1 the
%high-utilization demand rate
%dptim2 is the matrix addition to be used together with demrat2 the
%low-utilization demand rate
dpttim1 = dpttim11 + dpttim12 + dpttim13;
dpttim2 = dpttim21 + dpttim22 + dpttim23;
dptdemand1 = (itemnrts .* demrat1) .* dpttim1;
dptdemand2 = (itemnrts .* demrat2) .* dpttim2;

%-----------------------------------------------------------------
%dptdemand is the mean demand caused by the depot turnaround time.
dptdemand  = dptdemand1 + dptdemand2 ;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%demand is the Total Average Pipeline in according to Equation 27 of
%of Chapter III. It is used to compute the SMOEs and MMOEs.
```

```
demand = basdemand + dptdemand;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Calculate the Expected Number Of Backorders Ebi(t) for each LRU.

%make a matrix out of invlevel based each time intervals.
invmat = ones(timeint)' * invlevel;

%if demand is more than invmat, then there will be backorders
backord1 = demand - invmat;
%index to recognize those elements with backorders.

b1 = find(backord1 > 0.0);
inv1 = invmat(b1);
dem1 = demand(b1);
back1 = backord1(b1);
%-------------------------------------------------------------
%begin poisson computation by looping from zero to the maximum inventory
%level m.

m = max(inv1);

% this is the poisson term when k=0, or when the actual demand k is zero.
% poiterm is the number of backorders
%see Chapter 3, Section 1a for explanation why start from k = 0
poiterm = (inv1) ;
prob = poiterm;

for i = 1:m-1
ii = i * ones(dem1);

%the poi(k,x) is a function to compute the poisson probability that
%the actual number is k when the mean number is x. k and x can be
%in any form (scalar, vector or matrix).
probi = poi(ii,dem1);

%At each incremental loop, compute only those which inventory level is
%than or equal to i.
j = (inv1 >= i);

poit = j .* ((inv1 -ii).* probi);

%sum up all the poisson probabilities according to equation 3.35
```

```
poiterm = poiterm + poit;
prob = prob + probi;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% the computation of the expected backorder for each LRU.
% (see equation 3.35)
backorder = 0.0 * backord1;
backorder(b1) = (back1 + (poiterm .* exp(-dem1)) );

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% For Non-cannibalization policy

%calculate Total Expected Backorders (see equation 3.37).
totaleb = zeros(timeint)';
for i = 1:max(size(itemfr))
totaleb = totaleb + backorder(:,i);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% For Non-cannibalization policy

%Calculate operational availability (see equation 3.39).
av = 1 - (backorder/totalsys);
ao = ones(timeint)';

for i = 1:max(size(itemfr))
ao = ao .* av(:,i);
end

%------------------------------------------------------------
%Calculate ENMCS without cannibalization
%(see equation 3.48).

enmcs = totalsys * (1 - ao);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% For Cannibalization Policy

%Probability of shortages less than or equal to j
%(see Equations 3.43 and 3.44 of Chapter III).

%prshort is P(j): Probability of having less than or equal to
```

```
% j NMCS.  Form unity matrix for prshort from

prshort = ones(timeint)' * ones(1:totalsys+1);

for n = 0:totalsys
%if demand is more than invmat, then there will be backorders
backord3 = demand - invmat+n;
%index to recognize those elements with backorders.

b3 = find(backord3 > 0.0);
inv3 = invmat(b3);
dem3 = demand(b3);
%-------------------------------------------------------------------

m1 = max(inv3+n);

% start with poisson at k=0 as actual number and demand as mean number.
poij0 = ones(dem3);

for i = 1:m1
ii = i * ones(dem3);
%use the poisson function (matrix form)
poic = poi(ii,dem3);
j = (inv3+n >= i);

poic = j .* poic ;
poij0 = poij0 + poic;
end

probnbo = ones(timeint)' * ones(itemfr);
probnbo(b3) = poij0 .* exp(-dem3);

%prshort(:,1) can be interpreted as the probability of all
%systems availability

for j = 1:max(size(itemfr))
prshort(:,n+1) = prshort(:,n+1) .* probnbo(:,j);
end

end
%-------------------------------------------------------------------

%calculate Prob(ENMCS <= NMCS) (same result for non-cannibalization
```

```
%or cannibalization
pnmcsc = prshort(:,nmcs);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%calculate expected number of NMCS (ENMCSc) for cannibalization

enmcsc = zeros(timeint)';
for j = 1:totalsys+1;
enmcsc = enmcsc + (1 - prshort(:,j)) ;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%calculate operational availability with cannibalization
aoc = 1 - (enmcsc/totalsys);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function b = poi(k,x)
%poisson function that handles matrix, vector or scalar.
b = ((x.^k) ./ fact(k));
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function b = fact(k);
% factorial function that operates on matrix, vector or scalar.
m = max(k(:));
b = ones(k);
for i = 1:m
n = i * ones(k);
j = find(k >= i);
c = ones(k);
c(j) = n(j);
b = b .* c;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%END OF MATLAB PROGRAM
```

APPENDIX C. NUMERICAL RESULTS FROM ANALYTICAL VERSION OF THE WSM.

Tables shown below are numerical results from the Analytical Case One.

TABLE C-1 : ANALYTICAL RESULTS OF THE TOTAL PIPELINE FOR EACH LRU.

Time	LRU A	LRU B	LRU C	LRU D
0.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	0.0000000e+00
36.00	6.7392000e-02	6.9811200e-01	6.2208000e-02	1.9889280e+00
72.00	9.3000960e-02	9.5874048e-01	1.2441600e-01	3.9778560e+00
108.00	9.7718400e-02	1.0006272e+00	1.3747968e-01	5.1513235e+00
144.00	1.0243584e-01	1.0425139e+00	1.5054336e-01	6.3247910e+00
180.00	9.5921280e-02	9.6804864e-01	1.5323904e-01	7.1667706e+00
216.00	6.6942720e-02	6.6087936e-01	1.3519872e-01	7.3457741e+00
252.00	6.9301440e-02	6.8182272e-01	1.2534912e-01	7.6606877e+00
288.00	7.1660160e-02	7.0276608e-01	1.3188096e-01	8.2474214e+00
324.00	7.4018880e-02	7.2370944e-01	1.3841280e-01	8.8341552e+00
360.00	7.6377600e-02	7.4465280e-01	1.4494464e-01	9.4208890e+00
396.00	7.4018880e-02	7.2370944e-01	1.3841280e-01	8.8341552e+00
432.00	7.1660160e-02	7.0276608e-01	1.3188096e-01	8.2474214e+00
468.00	6.9301440e-02	6.8182272e-01	1.2534912e-01	7.6606877e+00
504.00	6.6942720e-02	6.6087936e-01	1.1881728e-01	7.0739539e+00
540.00	6.5370240e-02	6.4691712e-01	1.1446272e-01	6.6827981e+00
576.00	6.5370240e-02	6.4691712e-01	1.1446272e-01	6.6827981e+00
612.00	6.5370240e-02	6.4691712e-01	1.1446272e-01	6.6827981e+00
648.00	6.5370240e-02	6.4691712e-01	1.1446272e-01	6.6827981e+00
684.00	6.5370240e-02	6.4691712e-01	1.1446272e-01	6.6827981e+00
720.00	6.5370240e-02	6.4691712e-01	1.1446272e-01	6.6827981e+00

(TABLE C-1 cont..)

Time	LRU E	LRU F	LRU G
0.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
36.00	7.7587200e-01	1.0488960e+00	9.8496000e-02
72.00	1.0551859e+00	1.4264986e+00	1.9699200e-01
108.00	1.0862208e+00	1.4684544e+00	2.2063104e-01
144.00	1.1172557e+00	1.5104102e+00	2.4427008e-01
180.00	1.0189786e+00	1.3775501e+00	2.5149312e-01
216.00	6.6207744e-01	8.9505792e-01	2.2588416e-01
252.00	6.7759488e-01	9.1603584e-01	2.1275136e-01
288.00	6.9311232e-01	9.3701376e-01	2.2457088e-01
324.00	7.0862976e-01	9.5799168e-01	2.3639040e-01
360.00	7.2414720e-01	9.7896960e-01	2.4820992e-01
396.00	7.0862976e-01	9.5799168e-01	2.3639040e-01
432.00	6.9311232e-01	9.3701376e-01	2.2457088e-01
468.00	6.7759488e-01	9.1603584e-01	2.1275136e-01
504.00	6.6207744e-01	8.9505792e-01	2.0093184e-01
540.00	6.5173248e-01	8.8107264e-01	1.9305216e-01
576.00	6.5173248e-01	8.8107264e-01	1.9305216e-01
612.00	6.5173248e-01	8.8107264e-01	1.9305216e-01
648.00	6.5173248e-01	8.8107264e-01	1.9305216e-01
684.00	6.5173248e-01	8.8107264e-01	1.9305216e-01
720.00	6.5173248e-01	8.8107264e-01	1.9305216e-01
Time	**LRU H**	**LRU I**	**LRU J**
0.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
36.00	7.7414400e-01	2.9376000e+00	1.8869760e+00
72.00	1.0786406e+00	4.0343040e+00	2.8682035e+00
108.00	1.1483136e+00	4.2105600e+00	3.3965568e+00
144.00	1.2179866e+00	4.3868160e+00	3.9249101e+00
180.00	1.1586355e+00	4.0734720e+00	4.1387674e+00
216.00	8.4123648e-01	2.7809280e+00	3.7236326e+00
252.00	8.7607296e-01	2.8690560e+00	3.9878093e+00
288.00	9.1090944e-01	2.9571840e+00	4.2519859e+00
324.00	9.4574592e-01	3.0453120e+00	4.5161626e+00
360.00	9.8058240e-01	3.1334400e+00	4.7803392e+00
396.00	9.4574592e-01	3.0453120e+00	4.5161626e+00
432.00	9.1090944e-01	2.9571840e+00	4.2519859e+00
468.00	8.7607296e-01	2.8690560e+00	3.9878093e+00
504.00	8.4123648e-01	2.7809280e+00	3.7236326e+00
540.00	8.1801216e-01	2.7221760e+00	3.5475149e+00
576.00	8.1801216e-01	2.7221760e+00	3.5475149e+00
612.00	8.1801216e-01	2.7221760e+00	3.5475149e+00
648.00	8.1801216e-01	2.7221760e+00	3.5475149e+00
684.00	8.1801216e-01	2.7221760e+00	3.5475149e+00
720.00	8.1801216e-01	2.7221760e+00	3.5475149e+00

TABLE C-2 : ANALYTICAL RESULTS OF BACKORDERS FOR EACH LRU

Time	LRU A	LRU B	LRU C
0.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
36.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
72.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
108.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
144.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
180.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
216.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
252.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
288.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
324.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
360.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
396.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
432.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
468.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
504.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
540.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
576.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
612.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
648.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
684.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
720.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
Time	**LRU D**	**LRU E**	**LRU F**
0.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
36.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
72.00	0.0000000e+00	0.0000000e+00	0.0000000e+00
108.00	9.6399166e-01	0.0000000e+00	0.0000000e+00
144.00	1.7570721e+00	0.0000000e+00	0.0000000e+00
180.00	2.4317595e+00	0.0000000e+00	0.0000000e+00
216.00	2.5837480e+00	0.0000000e+00	0.0000000e+00
252.00	2.8571034e+00	0.0000000e+00	0.0000000e+00
288.00	3.3835732e+00	0.0000000e+00	0.0000000e+00
324.00	3.9275290e+00	0.0000000e+00	0.0000000e+00
360.00	4.4843017e+00	0.0000000e+00	0.0000000e+00
396.00	3.9275290e+00	0.0000000e+00	0.0000000e+00
432.00	3.3835732e+00	0.0000000e+00	0.0000000e+00
468.00	2.8571034e+00	0.0000000e+00	0.0000000e+00
504.00	2.3540075e+00	0.0000000e+00	0.0000000e+00
540.00	2.0350727e+00	0.0000000e+00	0.0000000e+00
576.00	2.0350727e+00	0.0000000e+00	0.0000000e+00
612.00	2.0350727e+00	0.0000000e+00	0.0000000e+00
648.00	2.0350727e+00	0.0000000e+00	0.0000000e+00
684.00	2.0350727e+00	0.0000000e+00	0.0000000e+00
720.00	2.0350727e+00	0.0000000e+00	0.0000000e+00

(TABLE C-2 cont..)

Time	LRU G	LRU H	LRU I	LRU J
0.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	0.0000000e+00
36.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	0.0000000e+00
72.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	1.1447217e+00
108.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	1.5772787e+00
144.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	2.0418910e+00
180.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	2.2366346e+00
216.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	1.8618360e+00
252.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	2.0988250e+00
288.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	2.3409888e+00
324.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	2.5873900e+00
360.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	2.8372476e+00
396.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	2.5873900e+00
432.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	2.3409888e+00
468.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	2.0988250e+00
504.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	1.8618360e+00
540.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	1.7072617e+00
576.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	1.7072617e+00
612.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	1.7072617e+00
648.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	1.7072617e+00
684.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	1.7072617e+00
720.00	0.0000000e+00	0.0000000e+00	0.0000000e+00	1.7072617e+00

APPENDIX D. NUMERICAL RESULTS FROM SIMULATION VERSION OF THE WSM.

Tables below show the numerical results for the case of Simulation Variation Four.

TABLE D-1 : SIMULATION RESULTS OF THE TOTAL PIPELINE FOR EACH LRU WITH STATISTICAL CONFIDENTIAL LIMITS

	LRU A at 95% C.I.			LRU B at 95% C.I.		
Sim.Time	Mean	LLimit	ULimit	Mean	LLimit	ULimit
0.00	0.000	0.000	0.000	0.000	0.000	0.000
36.00	0.051	0.045	0.057	0.515	0.494	0.535
72.00	0.081	0.073	0.089	0.831	0.804	0.857
108.00	0.111	0.101	0.120	1.128	1.097	1.159
144.00	0.142	0.131	0.153	1.381	1.347	1.414
180.00	0.174	0.162	0.185	1.589	1.553	1.625
216.00	0.196	0.184	0.209	1.639	1.604	1.675
252.00	0.217	0.204	0.230	1.625	1.591	1.660
288.00	0.227	0.214	0.240	1.512	1.479	1.545
324.00	0.228	0.215	0.241	1.362	1.329	1.394
360.00	0.216	0.203	0.228	1.178	1.148	1.209
396.00	0.194	0.182	0.206	1.011	0.982	1.040
432.00	0.176	0.165	0.187	0.873	0.845	0.900
468.00	0.162	0.152	0.173	0.785	0.759	0.812
504.00	0.152	0.141	0.162	0.701	0.676	0.726
540.00	0.139	0.129	0.149	0.625	0.602	0.649
576.00	0.129	0.120	0.139	0.591	0.569	0.614
612.00	0.121	0.112	0.131	0.557	0.535	0.579
648.00	0.108	0.099	0.118	0.538	0.516	0.560
684.00	0.102	0.093	0.112	0.536	0.514	0.558
720.00	0.097	0.088	0.106	0.536	0.514	0.557

(TABLE D-1 cont..)

	LRU C at 95% C.I.			LRU D at 95% C.I.		
0.00	0.000	0.000	0.000	0.000	0.000	0.000
36.00	0.053	0.046	0.059	1.810	1.772	1.848
72.00	0.093	0.084	0.101	3.448	3.396	3.500
108.00	0.126	0.116	0.136	4.967	4.904	5.029
144.00	0.158	0.147	0.169	6.344	6.274	6.413
180.00	0.197	0.184	0.209	7.424	7.350	7.498
216.00	0.226	0.213	0.239	7.952	7.879	8.026
252.00	0.252	0.238	0.265	8.216	8.143	8.288
288.00	0.270	0.256	0.284	8.194	8.122	8.266
324.00	0.274	0.260	0.287	7.960	7.889	8.031
360.00	0.273	0.259	0.287	7.503	7.434	7.572
396.00	0.265	0.251	0.279	6.991	6.922	7.060
432.00	0.249	0.236	0.262	6.449	6.381	6.517
468.00	0.232	0.219	0.245	6.020	5.953	6.086
504.00	0.216	0.203	0.228	5.687	5.622	5.753
540.00	0.199	0.186	0.211	5.389	5.324	5.454
576.00	0.192	0.181	0.204	5.234	5.169	5.299
612.00	0.179	0.167	0.190	5.141	5.076	5.205
648.00	0.168	0.157	0.179	5.093	5.029	5.158
684.00	0.166	0.155	0.177	5.072	5.007	5.137
720.00	0.154	0.143	0.165	5.074	5.009	5.139

	LRU E at 95% C.I.			LRU F at 95% C.I.		
0.00	0.000	0.000	0.000	0.000	0.000	0.000
36.00	0.555	0.534	0.576	0.786	0.760	0.811
72.00	0.924	0.896	0.952	1.288	1.254	1.321
108.00	1.246	1.214	1.278	1.693	1.655	1.732
144.00	1.525	1.489	1.560	2.046	2.004	2.089
180.00	1.740	1.703	1.778	2.315	2.271	2.360
216.00	1.770	1.733	1.806	2.334	2.290	2.377
252.00	1.715	1.680	1.751	2.198	2.156	2.240
288.00	1.562	1.529	1.596	1.983	1.942	2.023
324.00	1.393	1.360	1.426	1.729	1.690	1.767
360.00	1.209	1.177	1.240	1.512	1.476	1.549
396.00	1.027	0.997	1.057	1.301	1.266	1.336
432.00	0.897	0.869	0.925	1.146	1.113	1.180
468.00	0.776	0.749	0.803	1.011	0.980	1.043
504.00	0.694	0.668	0.719	0.927	0.897	0.957
540.00	0.625	0.601	0.649	0.860	0.831	0.889
576.00	0.600	0.577	0.624	0.811	0.784	0.839
612.00	0.591	0.568	0.615	0.798	0.771	0.825
648.00	0.587	0.565	0.610	0.785	0.759	0.811
684.00	0.583	0.561	0.605	0.760	0.734	0.786
720.00	0.560	0.538	0.581	0.742	0.716	0.768

(TABLE D-1 cont..)

	LRU G at 95% C.I.			LRU H at 95% C.I.		
0.00	0.000	0.000	0.000	0.000	0.000	0.000
36.00	0.085	0.077	0.093	0.587	0.566	0.609
72.00	0.154	0.143	0.166	0.971	0.942	1.000
108.00	0.216	0.203	0.230	1.320	1.287	1.352
144.00	0.263	0.248	0.277	1.607	1.571	1.643
180.00	0.314	0.298	0.329	1.840	1.802	1.878
216.00	0.358	0.341	0.374	1.918	1.880	1.956
252.00	0.400	0.382	0.417	1.865	1.828	1.902
288.00	0.420	0.402	0.437	1.746	1.710	1.782
324.00	0.421	0.404	0.439	1.567	1.532	1.602
360.00	0.409	0.391	0.426	1.396	1.363	1.429
396.00	0.399	0.383	0.416	1.222	1.190	1.253
432.00	0.371	0.355	0.387	1.061	1.031	1.091
468.00	0.349	0.333	0.364	0.919	0.891	0.948
504.00	0.320	0.305	0.335	0.832	0.805	0.859
540.00	0.298	0.284	0.313	0.749	0.723	0.775
576.00	0.282	0.268	0.296	0.714	0.689	0.739
612.00	0.268	0.254	0.282	0.697	0.672	0.721
648.00	0.260	0.246	0.274	0.671	0.647	0.695
684.00	0.252	0.239	0.266	0.657	0.634	0.681
720.00	0.245	0.231	0.258	0.648	0.624	0.672
	LRU I at 95% C.I.			**LRU J at 95% C.I.**		
0.00	0.000	0.000	0.000	0.000	0.000	0.000
36.00	2.204	2.162	2.246	1.482	1.447	1.516
72.00	3.579	3.521	3.636	2.668	2.621	2.714
108.00	4.766	4.697	4.835	3.685	3.630	3.740
144.00	5.759	5.683	5.834	4.583	4.523	4.644
180.00	6.358	6.279	6.436	5.262	5.199	5.324
216.00	6.046	5.968	6.124	5.404	5.342	5.465
252.00	5.509	5.434	5.585	5.312	5.252	5.373
288.00	4.991	4.918	5.065	5.108	5.049	5.168
324.00	4.497	4.426	4.568	4.767	4.710	4.825
360.00	4.006	3.937	4.075	4.400	4.344	4.457
396.00	3.561	3.496	3.626	4.031	3.976	4.086
432.00	3.239	3.177	3.301	3.679	3.625	3.733
468.00	2.971	2.911	3.031	3.394	3.340	3.447
504.00	2.760	2.703	2.817	3.183	3.131	3.235
540.00	2.585	2.530	2.639	3.015	2.964	3.066
576.00	2.470	2.417	2.522	2.899	2.849	2.949
612.00	2.428	2.377	2.479	2.828	2.778	2.878
648.00	2.355	2.306	2.404	2.789	2.740	2.838
684.00	2.343	2.295	2.391	2.775	2.727	2.824
720.00	2.347	2.299	2.395	2.783	2.734	2.832

TABLE D-2 : SIMULATION RESULTS OF THE BACKORDERS FOR EACH LRU WITH STATISTICAL CONFIDENCE LIMITS						
	LRU A at 95% C.I.			LRU B at 95% C.I.		
Sim.Time	Mean	LLimit	ULimit	Mean	LLimit	ULimit
0.00	0.000	0.000	0.000	0.000	0.000	0.000
36.00	0.000	0.000	0.000	0.001	0.000	0.003
72.00	0.002	0.001	0.004	0.013	0.009	0.018
108.00	0.006	0.004	0.008	0.041	0.034	0.048
144.00	0.008	0.006	0.011	0.074	0.065	0.084
180.00	0.012	0.009	0.016	0.113	0.100	0.125
216.00	0.016	0.013	0.020	0.115	0.103	0.127
252.00	0.019	0.015	0.023	0.103	0.091	0.114
288.00	0.018	0.014	0.022	0.083	0.073	0.093
324.00	0.016	0.013	0.020	0.061	0.053	0.070
360.00	0.013	0.010	0.016	0.042	0.035	0.049
396.00	0.009	0.006	0.011	0.029	0.023	0.034
432.00	0.006	0.004	0.008	0.021	0.016	0.026
468.00	0.006	0.003	0.008	0.015	0.011	0.019
504.00	0.006	0.004	0.009	0.012	0.009	0.016
540.00	0.004	0.002	0.006	0.009	0.006	0.012
576.00	0.004	0.002	0.006	0.006	0.003	0.008
612.00	0.005	0.003	0.007	0.005	0.003	0.007
648.00	0.005	0.003	0.007	0.004	0.002	0.006
684.00	0.006	0.004	0.009	0.006	0.003	0.009
720.00	0.005	0.003	0.008	0.004	0.002	0.006

(TABLE D-2 cont..)

	LRU C at 95% C.I.			LRU D at 95% C.I.		
0.00	0.000	0.000	0.000	0.000	0.000	0.000
36.00	0.002	0.000	0.003	0.015	0.011	0.020
72.00	0.004	0.003	0.006	0.233	0.213	0.253
108.00	0.007	0.005	0.010	0.852	0.813	0.892
144.00	0.008	0.005	0.011	1.771	1.716	1.826
180.00	0.013	0.010	0.017	2.645	2.580	2.709
216.00	0.016	0.012	0.020	3.084	3.016	3.151
252.00	0.018	0.014	0.022	3.316	3.248	3.384
288.00	0.021	0.017	0.026	3.289	3.222	3.356
324.00	0.021	0.017	0.025	3.074	3.008	3.139
360.00	0.021	0.017	0.025	2.662	2.600	2.725
396.00	0.021	0.017	0.026	2.233	2.174	2.292
432.00	0.019	0.015	0.023	1.814	1.759	1.869
468.00	0.015	0.011	0.018	1.503	1.452	1.554
504.00	0.013	0.010	0.016	1.283	1.236	1.330
540.00	0.012	0.009	0.015	1.108	1.064	1.153
576.00	0.010	0.007	0.013	1.026	0.984	1.069
612.00	0.007	0.005	0.009	0.963	0.921	1.005
648.00	0.007	0.004	0.009	0.938	0.896	0.979
684.00	0.007	0.005	0.010	0.940	0.899	0.982
720.00	0.008	0.005	0.010	0.941	0.899	0.982

	LRU E at 95% C.I.			LRU F at 95% C.I.		
0.00	0.000	0.000	0.000	0.000	0.000	0.000
36.00	0.023	0.018	0.028	0.001	0.000	0.000
72.00	0.091	0.081	0.102	0.017	0.013	0.021
108.00	0.187	0.171	0.202	0.053	0.044	0.061
144.00	0.306	0.286	0.326	0.103	0.090	0.115
180.00	0.403	0.380	0.426	0.156	0.140	0.171
216.00	0.394	0.372	0.416	0.142	0.128	0.157
252.00	0.356	0.335	0.377	0.112	0.099	0.125
288.00	0.281	0.262	0.299	0.088	0.077	0.099
324.00	0.230	0.213	0.246	0.056	0.048	0.065
360.00	0.170	0.155	0.184	0.038	0.031	0.044
396.00	0.129	0.116	0.142	0.025	0.019	0.030
432.00	0.099	0.088	0.110	0.020	0.015	0.025
468.00	0.075	0.065	0.085	0.013	0.009	0.017
504.00	0.057	0.048	0.066	0.014	0.010	0.019
540.00	0.045	0.038	0.053	0.010	0.006	0.014
576.00	0.040	0.033	0.047	0.007	0.004	0.010
612.00	0.039	0.032	0.046	0.005	0.002	0.007
648.00	0.036	0.030	0.043	0.003	0.001	0.005
684.00	0.030	0.024	0.036	0.003	0.002	0.005
720.00	0.027	0.021	0.033	0.003	0.001	0.005

(TABLE D-2 cont..)

	IRU G at 95% C.I.			IRU H at 95% C.I.		
0.00	0.000	0.000	0.000	0.000	0.000	0.000
36.00	0.003	0.001	0.005	0.023	0.019	0.028
72.00	0.014	0.011	0.018	0.110	0.098	0.121
108.00	0.024	0.019	0.029	0.212	0.195	0.228
144.00	0.031	0.026	0.036	0.332	0.311	0.353
180.00	0.044	0.037	0.050	0.450	0.426	0.475
216.00	0.049	0.042	0.056	0.474	0.449	0.499
252.00	0.057	0.050	0.064	0.430	0.406	0.454
288.00	0.061	0.053	0.069	0.376	0.354	0.398
324.00	0.061	0.053	0.068	0.303	0.283	0.323
360.00	0.055	0.047	0.062	0.236	0.219	0.252
396.00	0.048	0.041	0.055	0.180	0.166	0.195
432.00	0.040	0.034	0.046	0.127	0.114	0.139
468.00	0.032	0.026	0.037	0.102	0.091	0.113
504.00	0.025	0.021	0.030	0.081	0.071	0.091
540.00	0.024	0.019	0.028	0.065	0.056	0.074
576.00	0.021	0.016	0.025	0.060	0.051	0.069
612.00	0.023	0.018	0.027	0.051	0.044	0.059
648.00	0.023	0.018	0.027	0.043	0.036	0.049
684.00	0.024	0.019	0.028	0.045	0.038	0.052
720.00	0.023	0.018	0.027	0.040	0.033	0.047
	IRU I at 95% C.I.			IRU J at 95% C.I.		
0.00	0.000	0.000	0.000	0.000	0.000	0.000
36.00	0.011	0.007	0.015	0.277	0.258	0.296
72.00	0.161	0.144	0.178	0.999	0.962	1.035
108.00	0.502	0.470	0.535	1.840	1.791	1.890
144.00	0.954	0.909	1.000	2.655	2.598	2.712
180.00	1.287	1.233	1.340	3.300	3.239	3.360
216.00	1.124	1.074	1.174	3.431	3.371	3.491
252.00	0.848	0.804	0.891	3.341	3.282	3.400
288.00	0.632	0.594	0.669	3.138	3.080	3.196
324.00	0.467	0.434	0.499	2.810	2.754	2.866
360.00	0.339	0.312	0.366	2.464	2.410	2.517
396.00	0.235	0.212	0.257	2.128	2.076	2.179
432.00	0.175	0.155	0.194	1.813	1.765	1.862
468.00	0.138	0.121	0.156	1.584	1.537	1.632
504.00	0.105	0.090	0.120	1.411	1.367	1.456
540.00	0.084	0.070	0.097	1.283	1.241	1.325
576.00	0.065	0.053	0.077	1.194	1.153	1.235
612.00	0.061	0.050	0.073	1.140	1.099	1.180
648.00	0.046	0.036	0.056	1.104	1.065	1.144
684.00	0.038	0.029	0.046	1.101	1.062	1.140
720.00	0.037	0.028	0.046	1.111	1.072	1.151

APPENDIX E. OPUS-8 INPUT REQUIREMENTS AND RESULTS

A. OPUS-8 INPUT FORMAT AND REQUIREMENTS

```
*PROBLEM DEFINITION                                              rev.2
+--------------------------------------------------------------------+
!WRITE PROBLEM HEADER BELOW:                                         !
+--------------------------------------------------------------------+
! ANALYTICAL WSM USING OPUS-8 APPROXIMATION WITHOUT REPAIR           !
+--------------------------------------------------------------------+

+------------------------------------------------------------------+
! PROBLEM TYPE   (MARK YOUR CHOICE WITH AN "X"):                   !
+-+----------------------------------------------------------------+
!X! ANALYSIS OF A GIVEN ALLOCATION OF SPARES                       !
+-+----------------------------------------------------------------+
! ! INITIAL OR REPLENISHMENT PROCUREMENT OF SPARES                 !
+-+----------------------------------------------------------------+
! ! REALLOCATION OF A GIVEN ASSORTMENT OF SPARES                   !
+-+----------------------------------------------------------------+
! ! REALLOCATION FOLLOWED BY REPLENISHMENT PROCUREMENT             !
+-+----------------------------------------------------------------+

+------------------------------------------------------------------+
! INITIAL STOCK   (MARK YOUR CHOICE WITH AN "X"):                  !
+-+----------------------------------------------------------------+
! ! ZERO STOCK                                                     !
+-+----------------------------------------------------------------+
!X! A GIVEN ALLOCATION OF SPARES                                   !
+-+----------------------------------------------------------------+
! ! AVERAGE NUMBER OF DEMANDS DURING RESUPPLY TIME (OR TIME T)     !
+-+----------------------------------------------------------------+

+------------------------------------------------------------------+
!MEASURE OF EFFECTIVENESS (MOE)  (MARK YOUR CHOICE WITH AN "X")    !
+------------------------------------------------------------------+
!  STEADY STATE PROBLEMS:                                          !
+-+----------------------------------------------------------------+
! ! WAITING TIME  (WT)                                             !
+-+----------------------------------------------------------------+
! ! PROBABILITY OF MISSION SUCCESS    (PMS)                        !
+-+----------------------------------------------------------------+
!  ENDURANCE PROBLEMS:                                             !
+-+----------------------------------------------------------------+
! ! PROBABILITY OF NO BACKORDER DURING TIME PERIOD T      (PNBO)   !
+-+----------------------------------------------------------------+
! ! SUM OF BACKORDER TIME DURING TIME PERIOD T             (SBT)   !
+-+----------------------------------------------------------------+
!X! NUMBER OF BACKORDERS DURING TIME PERIOD T              (NBO)   !
+-+----------------------------------------------------------------+
```

*CONTROL PARAMETERS rev.3

```
+------------------------------------------------------------+
! LIMITS FOR INVESTMENT AND/OR MOE:              <DEFAULT>    !
+--------------+---------------------------------------------+
!              ! MINIMAL INVESTMENT              <  0.0  >    !
+--------------+---------------------------------------------+
!              ! MAXIMAL INVESTMENT              < 1E+18 >    !
+--------------+---------------------------------------------+
!              ! MINIMAL MOE                     <MOE DEP.>   !
+--------------+---------------------------------------------+
!              ! MAXIMAL MOE                     <MOE DEP.>   !
+--------------+---------------------------------------------+
!              ! NUMBER OF POINTS IN FINAL CURVE  < 30 >      !
+--------------+---------------------------------------------+

+------------------------------------------------------------+
! FORMAT FOR ALL OUTPUT:                           <DEFAULT> !
+----------+-------------------------------------------------+
!          ! NUMBER OF LINES PER PAGE              <  64   > !
+----------+-------------------------------------------------+
!          ! NUMBER OF CHARACTERS PER LINE         <  80   > !
+----------+-------------------------------------------------+

+-------------------------------------------------------------------+
! USE OF THE COMPOUND POISSON DISTRIBUTION IN THE CALCULATIONS:      !
! (MARK SELECTED MODE WITH "X")         (VMR = VARIANCE / MEAN)     !
+-+-----------------------------------------------------------------+
! ! AUTOMATIC COMPOUND POISSON TO HANDLE WAITING TIME VARIANCE      !
+-+-----------------------------------------------------------------+
!X! STANDARD POISSON DISTRIBUTION (VMR = 1 FOR DEMAND AT ALL STATIONS) !
+-+-----------------------------------------------------------------+
! ! COMPOUND POISSON DEMANDS (VMR VALUE ENTERED EXPLICITLY BELOW)   !
+-+-----------------------------------------------------------------+
  !          ! VMR FOR DEMAND AT STATIONS DIRECTLY SERVING SYSTEMS  !
  +----------+------------------------------------------------------+

+-+----------------------------------------------------------+
! ! EXCLUDE INITIAL STOCK IN RESULTS          ("X" IF YES)    !
+-+----------------------------------------------------------+

+------------------------------------------------------------+
!MULTIPLE REMOVAL HANDLING     (MARK YOUR CHOICE WITH AN "X"): !
!(ONLY RELEVANT IF REMOVAL RATE FACTOR EXCEEDS 1 FOR SOME ITEM)!
+-+----------------------------------------------------------+
! ! POSSIBLE MULTIPLE REMOVALS INCLUDED IN GIVEN REPAIR TIMES  !
+-+----------------------------------------------------------+
! ! POSSIBLE MULTIPLE REMOVALS WILL INCREASE GIVEN REPAIR TIMES!
+-+----------------------------------------------------------+
! ! CREATE INPUT FILE TO PROGRAM "OPUS-8W"       ("X" IF YES) !
+-+----+------------------------------+---------------------+
       !                              ! FILE NAME  <"infile".W>!
       +------------+-----------------+---------------------+
       !            ! OPUS 8W GROUP IDENTIFICATION             !
       +------------+------------------------------------------+
```

*STATION DATA

(1) ID TAG	(2) DENOMINATION	(3) QUAN-TITY PER SUPPOR-TING STA-TION(S)	(4) SUB SET OF STA-TION (*)	(5) GROUP OF STA-TIONS
DEFAULT:		1	-	-
LOC1	DEMAND GEN ST	1		

*SYSTEM DATA

(1) SYSTEM ID TAG	(2) DEPLOYED AT FOLLOWING STATION(S)	(3) QUAN-TITY	(4) UTILIZA-TION PER CALENDAR TIME	(5) MEAN TIME TO REPAIR
DEFAULT:		-	1.0	0.0
SYSTEM	LOC1	24	0.1467	2.0

*ITEM DATA

(1) ID TAG	(2) DENOMINATION	(3) UNIT PRICE	(4) FAILURE RATE 10 TO-6	(5) ITEM (REPAIR AND STOCK) CATEGORY
DEFAULT:		-	-	<ITEM ID>
LRU01	A	1000.	390.	
LRU02	B	1000.	4040.	
LRU03	C	1000.	360.	
LRU04	D	1000.	11510.	
LRU05	E	1000.	4490.	
LRU06	F	1000.	6070.	
LRU07	G	1000.	570.	
LRU08	H	1000.	4480.	
LRU09	I	1000.	17000.	
LRU10	J	1000.	10920.	

*ITEM STRUCTURE rev.2

(1) ITEM ID TAG	(2) MOTHER ITEM OR SYSTEM (ID TAG)	(3) QUANTITY	(4) REMOVAL RATE FACTOR (*)	(5) ENVIRONMENT FACTOR (*)
DEFAULT:		1	1.0	1.0
LRU01	SYSTEM	1	1.0	
LRU02	SYSTEM	1	1.0	
LRU03	SYSTEM	1	1.0	
LRU04	SYSTEM	1	1.0	
LRU05	SYSTEM	1	1.0	
LRU06	SYSTEM	1	1.0	
LRU07	SYSTEM	1	1.0	
LRU08	SYSTEM	1	1.0	
LRU09	SYSTEM	1	1.0	
LRU10	SYSTEM	1	1.0	

*STOCK POLICY AND TIME PERIOD

TO BE USED IN ENDURANCE PROBLEMS			
(1) ITEM CATEGORY	(2) STOCKED AT FOLLOWING (GROUP OF) STATIONS	(3) INITIAL STOCK LEVEL	(4) TIME PERIOD T
DEFAULT:		0	-
LRU01	LOC1	1	360.
LRU02	LOC1	3	360.
LRU03	LOC1	1	360.
LRU04	LOC1	5	360.
LRU05	LOC1	2	360.
LRU06	LOC1	4	360.
LRU07	LOC1	1	360.
LRU08	LOC1	2	360.
LRU09	LOC1	6	360.
LRU10	LOC1	2	360.

*ALLOCATION OF SPARES rev.2

STOCK LEVELS PER ITEM AND STATION											
(1) ITEM ID TAG	(2) TOTAL NUMBER	(3) TOTAL NUMBER OF EACH STATION (4) STATION ID TAG									
/ / / / / / / / /	/ / / / / / / / /	1									
/ / / / / / / / /	/ / / / / / / / /	LOC1									
LRU01		1									
LRU02		3									
LRU03		1									
LRU04		5									
LRU05		2									
LRU06		4									
LRU07		1									
LRU08		2									
LRU09		6									
LRU10		2									

*OUTPUT SELECTION rev.2

MARK WITH "/" TO SUPPRESS OUTPUT (BLANK MEANS YES)	
/	ID: INPUT DATA
	ID.1 INPUT FORMS
	ID.2 RESTRUCTURED INPUT DATA
	ID.2.1 INITIAL STOCK LEVELS
	ID.2.2 DEPLOYMENT OF SYSTEMS
	ID.2.3 UTILIZATION OF SYSTEMS
	ID.3 SIGNIFICANCE LEVEL DATA
	IR: INTERMEDIATE RESULTS
	IR.1 SYSTEM SUMMARY DATA
	IR.2 ITEM AND STATION RELATED VARIABLES

```
    ! ! IR.2.1   DEMAND RATE                                            !
    +-+                                                                 !
    !/! IR.2.2   RESUPPLY TIME GIVEN NO SHORTAGE (OR TIME PERIOD T)     !
    +-+                                                                 !
    !/! IR.2.3   THE COMPONENTS OF RESUPPLY TIME                        !
    +-+                                                                 !
    ! ! IR.2.4   AVERAGE NUMBER OF DEMANDS                              !
    +-+                                                                 !
    ! ! IR.2.5   INITIAL ALLOCATION OF SPARES                           !
    +-+                                                                 !
    !/! IR.2.6   SIGNIFICANCE LEVELS                                    !
  +-+-+-----------------------------------------------------------------+
  !/! IR.3   SYSTEM AND DGS RELATED VARIABLES                           !
  +-+-+-----------------------------------------------------------------+
    ! ! IR.3.1   MDT GIVEN NO SHORTAGE                                  !
  +-+-+-----------------------------------------------------------------+
  !/!IR.4  SIGNIFICANCE LEVEL DATA                                      !
  +-+-+-----------------------------------------------------------------+
    ! ! IR.4.1   NUMBER OF POINTS                                       !
    +-+-----------------------------------------------------------------+
    ! ! IR.4.2   PRINTING CALCULATED POINTS OF THE C/E-CURVE(S)         !
    +-+-+---------------------------------------------------------------+
      ! ! IR.4.2.1   ALL CALCULATED POINTS                              !
      +-+                                                               !
      ! ! IR.4.2.2   ALL OPTIMAL POINTS                                 !
      +-+                                                               !
      ! ! IR.4.2.3   ALL SELECTED POINTS                                !
    +-+-+---------------------------------------------------------------+
    ! ! IR.4.3   PLOTTING CALCULATED POINTS OF THE C/E-CURVE(S)         !
    +-+-+---------------------------------------------------------------+
      ! ! IR.4.3.1   ALL CALCULATED POINTS                              !
      +-+                                                               !
      ! ! IR.4.3.2   ALL OPTIMAL POINTS                                 !
      +-+                                                               !
      ! ! IR.4.3.3   ALL SELECTED POINTS                                !
+-+-+-+-+---------------------------------------------------------------+
! ! FR:  FINAL RESULTS                                                  !
+-+-+-------------------------------------------------------------------+
  ! ! FR.1   RESULTS PER POINT (IN FOLLOWING INTERVAL IF NOT ALL POINTS)!
  +-+--------------------------------+----------+-----------------------+
                                     !          ! INVEST. LOWER LIMIT   !
                                     +----------+-----------------------+
                                     !          ! INVEST. UPPER LIMIT   !
    +-+------------------------------+----------+-----------------------+
    !/! FR.1.1   TOTAL STOCK LISTING                                    !
    +-+-----------------------------------------------------------------+
    ! ! FR.1.2   ITEM AND STATION RELATED VARIABLES                     !
    +-+-+---------------------------------------------------------------+
      ! ! FR.1.2.1   ALLOCATION OF SPARES                               !
      +-+                                                               !
```

```
   !/! FR.1.2.2  VALUE OF OPTIMIZATION MOE                           !
   +-+                                                               !
   !/! FR.1.2.3  EXPECTED SUM OF BACKORDERTIME DURING TIME PERIOD T  !
   +-+                                                               !
   !/! FR.1.2.4  PROBABILITY OF BACKORDER DURING TIME PERIOD T       !
   +-+                                                               !
   ! ! FR.1.2.5  EXPECTED NUMBER OF BACKORDERS  (*)                  !
   +-+                                                               !
   !/! FR.1.2.6  RISK OF SHORTAGE  (*)                               !
   +-+                                                               !
   !/! FR.1.2.7  TOTAL RESUPPLY TIME INCL. WAITING TIME  (*)         !
   +-+                                                               !
   !/! FR.1.2.8  INVESTMENT                                          !
 +-+-+-------------------------------------------------------------+
 ! ! FR.1.3  SYSTEM AND DGS RELATED VARIABLES                        !
 +-+-+-------------------------------------------------------------+
   !/! FR.1.3.1  VALUE OF OPTIMIZATION MOE                           !
   +-+                                                               !
   !/! FR.1.3.2  OPERATIONAL AVAILABILITY  (*)                       !
   +-+                                                               !
   ! ! FR.1.3.3  EXPECTED NUMBER OF NOR    (*)                       !
   +-+                                                               !
   ! ! FR.1.3.4  EXPECTED NUMBER OF BACKORDERS  (*)                  !
   +-+                                                               !
   !/! FR.1.3.5  RISK OF SHORTAGE  (*)                               !
   +-+                                                               !
   !/! FR.1.3.6  TOTAL MEAN DOWN TIME INCL. WAITING TIME  (*)        !
 +-+-+-------------------------------------------------------------+
 ! ! FR.1.4  MEAN VALUES PER ITEM FOR EACH RELEVANT MEASURE          !
 +-+---------------------------------------------------------------+
 ! ! FR.1.5  MEAN VALUES PER STATION FOR EACH RELEVANT MEASURE       !
+-+-+--------------------------------------------------------------+
! ! FR.2  PRINTING FINAL C/E-CURVE FOR INV. VS ALL RELEVANT MEASURES !
+-+----------------------------------------------------------------+
! ! FR.3  TRACEBACK OF FINAL POINTS TO INTERMEDIATE CURVES           !
+-+----------------------------------------------------------------+
! ! FR.4  MARGINAL COST EFFECTIVENESS                                !
+-+-+--------------------------------------------------------------+
 !/! FR.4.1  FOR INVESTMENT VERSUS WAITING TIME(*)                   !
 +-+                                                                 !
 ! ! FR.4.2  "     "     "   EXPECTED NUMBERS OF BACKORDERS(*)       !
 +-+                                                                 !
 ! ! FR.4.3  "     "     "   N O R                          (*)      !
 +-+---------------------------------------------------------------+
```

B. RESULTS USING OPUS-8'S ENDURANCE OPTION TO APPROXIMATE WSM

OPUS 8 (rev 3.2) MINDEF Singapore
ANALYTICAL WSM USING OPUS-8 APPROXIMATION WITHOUT REPAIR
JAN 08 1991 15:45:29

I N T E R M E D I A T E   R E S U L T S

Number of different items: 10
Number of different stations: 1
Number of stock positions: 10

Number of individual systems: 24
Operating time per system: 360.0
Total operating time: 8640.

NOTE! The figures above do not consider utilization factors.
All "Sum Of Backorder Time" values calculated by OPUS-8 must
be compared against operating times as presented above!

Investment of initial stock: 27000.0

*SYSTEM SUMMARY DATA

(1) SYSTEM ID TAG	(2) TOTAL FAILURE RATE (10 TO-6)	(3) TOTAL ITEM PRICE	(4) MTBF	(5) DEMAND FLOW (10 TO-6)	(6) ASYMP-TOTIC AVAIL-ABILITY
SYSTEM	59830.0	10000.0	16.71	8777.1	

*ALLOCATION OF SPARES

STOCK LEVELS PER ITEM AND STATION											
(1) ITEM ID TAG	(2) TOTAL NUM-BER	(3) TOTAL NUMBER OF EACH STATION / (4) STATION ID TAG									
		1									
		LOC1									
LRU01	1	1									
LRU02	3	3									
LRU03	1	1									
LRU04	5	5									
LRU05	2	2									
LRU06	4	4									
LRU07	1	1									
LRU08	2	2									
LRU09	6	6									
LRU10	2	2									

*DEMAND RATE

TOTAL DEMAND RATE PER ITEM AND STATION (10 TO-6)						
(1) ITEM ID TAG	(2) TOTAL NUMBER OF EACH STATION / (3) STATION ID TAG					
	1					
	LOC1					
LRU01	1373.11					
LRU02	14224.0					
LRU03	1267.49					
LRU04	40524.4					
LRU05	15808.4					
LRU06	21371.3					
LRU07	2006.86					
LRU08	15773.2					
LRU09	59853.6					
LRU10	38447.1					

*AVERAGE NUMBER OF DEMANDS

AVERAGE NUMBER OF DEMANDS DURING RESUPPLY TIME / TIME PERIOD T						
(1) ITEM ID TAG	(2) TOTAL NUMBER OF EACH STATION					
	(3) STATION ID TAG					
/ / / / / /	1					
/ / / / / /	LOC1					
LRU01	0.49					
LRU02	5.12					
LRU03	0.46					
LRU04	14.59					
LRU05	5.69					
LRU06	7.69					
LRU07	0.72					
LRU08	5.68					
LRU09	21.55					
LRU10	13.84					

F I N A L   R E S U L T S

*POINT NO. 1

INVESTMENT..................	27000.00
SUM OF BACKORDERTIME.........	6805.2935
AVERAGE NORS DURING T........	18.9036
ENDURANCE FACTOR.............	
NUMBER OF BACKORDERS.........	50.8480
PROB. OF NO BACKORDER........	0.000000
PROB. AT LEAST ONE BACKORDER.	1.000000

*ALLOCATION OF SPARES

STOCK LEVELS PER ITEM AND STATION											
(1) ITEM ID TAG	(2) TOTAL NUMBER	(3) TOTAL NUMBER OF EACH STATION / (4) STATION ID TAG									
		1									
		LOC1									
LRU01	1	1									
LRU02	3	3									
LRU03	1	1									
LRU04	5	5									
LRU05	2	2									
LRU06	4	4									
LRU07	1	1									
LRU08	2	2									
LRU09	6	6									
LRU10	2	2									

*EXPECTED NUMBER OF BACKORDERS /ITEM & STATION

EXPECTED NUMBER OF BACKORDERS PER ITEM AND STATION						
(1) ITEM ID TAG	(2) TOTAL NUMBER OF EACH STATION / (3) STATION ID TAG					
	1					
	LOC1					
LRU01	0.1043					
LRU02	2.2780					
LRU03	0.0899					
LRU04	9.5903					
LRU05	3.7170					
LRU06	3.7676					
LRU07	0.2080					
LRU08	3.7046					
LRU09	15.5473					
LRU10	11.8410					

*ENDURANCE MEASURES /ITEM

(1) ITEM ID TAG	(2) EXPECTED SUM OF BACKORDER TIME	(3) PROBA- BILITY OF NO BACKORDER	(4) EXPECTED NUMBER OF BACK- ORDERS	(5) INVEST- MENT
LRU01	13.0	0.911514	0.1043	1000.0
LRU02	249.1	0.248497	2.2780	3000.0
LRU03	11.2	0.922747	0.0899	1000.0
LRU04	1196.1	0.003708	9.5903	5000.0
LRU05	492.3	0.077263	3.7170	2000.0
LRU06	408.1	0.118566	3.7676	4000.0
LRU07	26.4	0.836348	0.2080	1000.0
LRU08	490.4	0.077958	3.7046	2000.0
LRU09	2069.4	0.000083	15.5473	6000.0
LRU10	1849.4	0.000108	11.8410	2000.0

*ENDURANCE MEASURES /STATION

(1) STA- TION ID TAG	(2) EXPECTED SUM OF BACKORDER TIME	(3) PROBA- BILITY OF NO BACKORDER	(4) EXPECTED NUMBER OF BACK- ORDERS	(5) INVEST- MENT
LOC1	6805.3	0.000000	50.8480	27000.0

*MARGINAL COST EFFECTIVENESS (AVERAGE NORS)

ME/C IS THE CHANGE IN AVERAGE NORS (AVERAGE OVER TIME PERIOD T) ACHIEVED BY A UNIT INCREASE IN INVESTMENT.
MC/E IS THE CHANGE IN INVESTMENT ACHIEVED BY ALLOWING NORS TO INCREASE ONE UNIT.

ME/C = 1/(MC/E)

(1) POINT NO.	(2) INVEST- MENT	(3) EXPECTED SUM OF BACKORDER TIME	(4) AVERAGE NORS (DURING TIME PERIOD T)	(5) MARGINAL COST PER EFFEC- TIVENESS (MC/E)	(6) MARGINAL EFFEC- TIVENESS PER COST (ME/C)
1	27000.0	6805.3	18.9036		

*MARGINAL COST EFFECTIVENESS (BACKORDERS)

ME/C IS THE CHANGE IN NO. OF BACKORDERS ACHIEVED BY A UNIT INCREASE IN INVESTMENT.
MC/E IS THE CHANGE IN INVESTMENT ACHIEVED BY ALLOWING THE NO. OF BO TO INCREASE ONE UNIT.

ME/C = 1/(MC/E)

(1) POINT NO.	(2) INVESTMENT	(3) EXPECTED NUMBER OF BACKORDERS	(4) MARGINAL COST PER EFFECTIVENESS (MC/E)	(5) MARGINAL EFFECTIVENESS PER COST (ME/C)
1	27000.0	50.8480		

*FINAL C/E-CURVE (SBT)

(1) POINT NO.	(2) INVESTMENT	(3) EXPECTED SUM OF BACKORDER TIME	(4) ENDURANCE FACTOR	(5) PROBABILITY OF NO BACKORDER	(6) PROBABILITY OF AT LEAST ONE BACKORDER	(7) EXPECTED NUMBER OF BACKORDERS
1	27000.0	6805.3		0.000000	1.000000	50.8480

END OF OUTPUT ______ OPUS 8

LIST OF REFERENCES

1. Eriksson B., OPUS8 - A Model For Optimal Allocation of Spares, Systecon AB, Stockholm, Sweden, March 90.

2. Isaacson K. E. and Boren P., "Dyna-METRIC Version 5 : An Assessment Model that Relates Logistics Resources and Policies to Wartime Readiness", R-3612-F, RAND Corporation, August 1988.

3. Slay F. M. and King R. M.," Prototype Aircraft Sustainability Model: An Item-specific Inventory Model that Relates Investment of Spares to Aircraft Sorties Generation Capability During a Conflict", AF601R2, Logistics Management Institute, March 1987.

4. CACI Products Company, , "A Discrete-event Object-oriented Simulation Software", MODSIM II Manuals, 1989.

5. The MathWorks, Inc., "An Interactive Software Package for Scientific and Engineering Numerical Computation", PC-MATLAB User Guide, 1988.

6. King R. M. and Mattern V. A., "War Reserve Materiel Policy Issues", Briefing Book AF902B1, Logistic Management Institute, November 1989.

7. Ross, S. M., Introduction to Probability Models, Academic Press, 1985.

8. Takacs, L., Introduction to the Theory of Queues, Oxford University Press, New York, 1962.

9. Hillestad R. J. and Carrillo M. J., "Models and Techniques for Recoverable Item Stockage When Demand and the Repair Process are Nonstationary-Part I: Performance Measurement", N-1482-AF, The RAND Corporation, May 1980.

10. Ross, S. M., Stochastic Processes, John Wiley & Sons, 1985.

11. Feeney G. J. and Sherbrooke C. C., " The (s-1, s) Inventory Policy Under Compound Poisson Demand: A Theory of Recoverable Item Stockage", RM-4176-PR, RAND Corporation, September 1964.

12. Feller, W. An introduction to Probability Theory and Its Application, Vol. 1, Wiley Series, New York, 1968.

13. Muckstadt J. A, "A Model for Multi-Echelon Multi-Indenture Inventory Problems", AD-Technical Report TR 79-1, US Army Inventory Research Office, December 1978.

14. Sherbrooke, C. C., "Discrete Compound Poisson Processes and Tables of the Geometric Poisson Distribution", The RAND Corporation, RM-4831-PR, July 1966.

15. Sherbrooke C. C., "METRIC: A Multi-echelon Technique for Recoverable Item Control", Operations Research, Vol 16, p. 122-141, 1986.

16. Hillestad R. J., "Dyna-Metric: Dynamic Multi-Echelon Technique for Recoverable Item Control", RAND Corporation, R-2785-AF, July 1982.

17. STSC Inc., "An Interactive Graphical Statistical Software Package", STATGRAPHICS 4.0 User Guide, 1989.

18. Banks J. and Carson J. S., Discrete-event System Simulation, Prentice-Hall International Series, 1984.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Defense Logistics Studies Information Exchange U.S. Army Logistics Management Center Alexandria, VA 22304-6145	1
3.	Library, Code 52 Naval Postgraduate School Monterey, CA 93943-5002	2
4.	Professor Emeritus M. B. Kline, Code AS/Kx Department of Administrative Science Naval Postgraduate School Monterey, CA 93943-5000	1
5.	Associate Professor A. W. McMasters, Code AS/Mg Department of Administrative Science Naval Postgraduate School Monterey, CA 93943-5000	1
6.	Assistant Professor M. P. Bailey, Code OR/Ba Department of Operations Research Naval Postgraduate School Monterey, CA 93943-5000	1
7.	Chief Defence Scientist MINDEF Building Gombak Drive Off Bukit Timah Road, SE2366 Republic of Singapore	1

8. Deputy Secretary (Technology) 1
MINDEF Building
Gombak Drive
Off Bukit Timah Road, SE2366
Republic of Singapore

9. Director 1
Defence Material Organisation
LEO Building
Paya Lebar Airport, SE1953
Republic of Singapore

10. Head, Operations Analysis Department 1
MINDEF Building
Gombak Drive
Off Bukit Timah Road, SE2366
Republic of Singapore

11. Lim Hung Heng 2
Defence Materials Organisation
Reliability Technology
LEO Building
Paya Lebar Airport, SE1953
Republic of Singapore

12. Olof Wååk 1
Systecon AB
S-111 43 Stockholm
Sweden

END